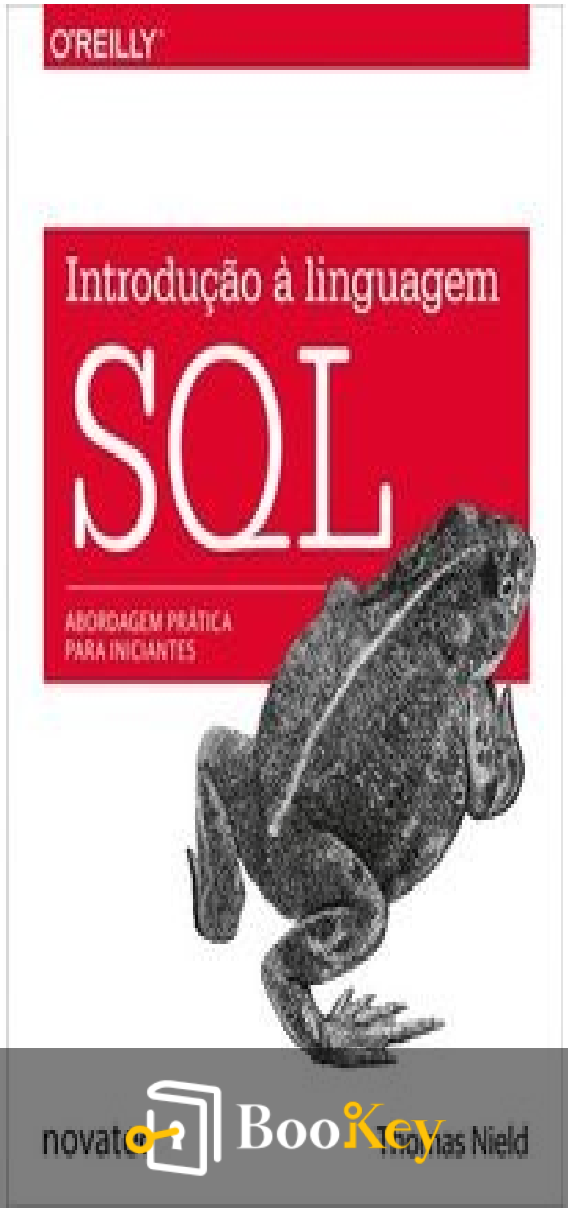
Introduçao a Linguagem SQL PDF

Thomas Nield



Mais Livros Grátis
Escaneie para baixar
Ouça-o

Introduçao a Linguagem SQL

Domine os Fundamentos de SQL para Transformar suas Habilidades em Gestão de Dados.

Escrito por Bookey

Saiba mais sobre o resumo de Introduçao a Linguagem SQL

Ouvir Introduçao a Linguagem SQL Audiolivro

Mais Livros Grátis

Escaneie para baixar

Ouça-o

Sobre o livro

No mundo orientado a dados de hoje, as empresas acumulam vastas quantidades de informações, mas muitas não possuem as habilidades necessárias para aproveitá-las de forma eficaz. "Introdução à Linguagem SQL", de Thomas Nield, é um guia prático e conciso, desenvolvido para profissionais de negócios e de TI que desejam dominar o SQL. Este recurso prático não apenas aborda os fundamentos essenciais da linguagem, mas também capacita você a criar e gerenciar seus próprios bancos de dados. Com exercícios que permitem praticar suas habilidades de forma independente—sem a necessidade de um servidor de banco de dados—você aprenderá a navegar por comandos SQL importantes para recuperar e manipular dados de forma eficiente. O livro também oferece estratégias para desenhar bancos de dados adaptados às suas necessidades específicas, tornando-se uma ferramenta valiosa para aprimorar suas capacidades de gestão de dados.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Sobre o autor

Thomas Nield é um profissional e educador apaixonado por dados, conhecido por sua expertise em SQL e análises de dados. Com uma formação em ciência da computação e negócios, Nield combina proficiência técnica com aplicação prática, tornando conceitos complexos acessíveis a aprendizes de todos os níveis. Ele é autor de diversos recursos que buscam desmistificar a manipulação e análise de dados, focando em abordagens práticas e hands-on que capacitam os leitores a desenvolver suas habilidades de forma eficaz. Como um defensor dedicado da alfabetização em dados, Nield contribui para a comunidade mais ampla por meio de workshops, cursos online e trabalhos escritos, ajudando indivíduos a aproveitar o poder dos dados para gerar insights e tomar decisões informadas.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Ad

Escanear para baixar

Experimente o aplicativo Bookey para ler mais de 1000 resumos dos melhores livros do mundo

Desbloqueie 1000+ títulos, 80+ tópicos

Novos títulos adicionados toda semana

duct & Brand | Liderança & Colaboração | Gerenciamento de Tempo | Relacionamento & Comunica

Estratégia de Negócios | Criatividade | Memórias | Conheça a Si Mesmo | Psicologia Po

mpreendedorismo | História Mundial | Comunicação entre Pais e Filhos | Autocuidado | Mi

Visões dos melhores livros do mundo

Teste gratuito com Bookey

Lista de conteúdo do resumo

Capítulo 1 : 1. Por Que Aprender SQL?

Capítulo 2 : 2. Bancos de Dados

Capítulo 3 : 3. SQLite

Capítulo 4 : 4. SELECT

Capítulo 5 : 5. WHERE

Capítulo 6 : 6. GROUP BY e ORDER BY

Capítulo 7 : 7. Instruções CASE

Capítulo 8 : 8. JOIN

Capítulo 9 : 9. Design de Banco de Dados

Capítulo 10 : 10. Gerenciando Dados

Capítulo 11 : 11. Indo em Frente

Capítulo 12 : A. Operadores e Funções

Capítulo 13 : B. Tópicos Suplementares

Mais Livros Grátis

Escaneie para baixar

Ouça-o

Capítulo 1 Resumo : 1. Por Que Aprender SQL?

Capítulo 1: Por Que Aprender SQL?

O Que É SQL e Por Que É Valioso no Mercado?

O cenário empresarial está mudando rapidamente devido à tecnologia e ao aumento dos dados. As empresas estão investindo pesadamente na aquisição e armazenamento de dados, mas enfrentam dificuldades para utilizar esses dados de maneira eficaz. SQL (Structured Query Language) é essencial para acessar e manipular dados e extrair insights empresariais. A demanda por habilidades em SQL está em

Mais Livros Grátis
Escaneie para baixar
Ouça-o

crescimento, já que muitas organizações coletam dados, mas não têm pessoal capacitado para analisá-los. Uma pesquisa realizada em 2015 pela Stack Overflow destacou a alta valorização do SQL e os salários competitivos.

Para Quem É SQL?

Há uma ideia comum de que SQL é uma habilidade exclusiva de TI. Na realidade, SQL é benéfico para profissionais de negócios, gerentes, profissionais de TI e engenheiros, oferecendo diversas oportunidades de carreira. Entender SQL permite que os indivíduos obtenham insights mais profundos sobre seus negócios, abrindo caminhos para funções analíticas, gerenciais, estratégicas ou orientadas a projetos nas áreas de negócios e TI.

Capítulo 2: Bancos de Dados

O Que É um Banco de Dados?

Um banco de dados é essencialmente um sistema para coletar e organizar dados. Embora termos gerais como planilhas ou arquivos de texto simples possam se qualificar, um banco de

Mais Livros Grátis
Escaneie para baixar
Ouça-o

dados profissional geralmente se refere a um sistema de gerenciamento de banco de dados relacional (RDBMS), que organiza dados em tabelas com relacionamentos.

Explorando Bancos de Dados Relacionais

Bancos de dados consistem em tabelas com linhas e colunas, semelhantes a planilhas, onde relacionamentos entre tabelas (como PEDIDO e CLIENTE) armazenam informações de forma concisa. Essa estrutura melhora a normalização dos dados, reduzindo redundâncias e melhorando a manutenção.

Por Que Separar Tabelas?

A normalização separa os dados em tabelas distintas, prevenindo redundâncias e simplificando a gestão dos dados. Por exemplo, ao invés de armazenar informações de clientes em uma tabela de PEDIDO, elas são armazenadas em uma tabela separada de CLIENTE para maior eficiência.

Escolhendo Uma Solução de Banco de Dados

As soluções de banco de dados podem ser classificadas como leves ou centralizadas. Bancos de dados leves, como SQLite

Mais Livros Grátis
Escaneie para baixar
Ouça-o

e Microsoft Access, atendem pequenos grupos e têm pouca sobrecarga. Bancos de dados centralizados, como MySQL ou Oracle, suportam bases de usuários e aplicações maiores, operando em servidores para gerenciar ambientes de alta demanda de forma eficiente.

Bancos de Dados Leves

Bancos de dados leves são fáceis de usar e ideais para projetos individuais ou de pequenas equipes. O SQLite é recomendado para aprender SQL devido à sua configuração e uso simples, enquanto o Microsoft Access oferece recursos empresariais robustos.

Bancos de Dados Centralizados

Para grupos de usuários maiores, bancos de dados centralizados são necessários. Eles operam em servidores para gerenciar tráfego e requisições significativas, oferecendo diversas soluções. O SQL permanece consistente em diferentes plataformas, permitindo que os usuários interajam de forma uniforme com os dados, independentemente da solução de banco de dados específica utilizada.
Em resumo, entender os fundamentos de SQL e bancos de

Mais Livros Grátis
Escaneie para baixar
Ouça-o

dados não só melhora as habilidades de gestão de dados, mas também abre inúmeras oportunidades de carreira em diversos setores.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Exemplo

Ponto chave:A demanda por habilidades em SQL vai além dos profissionais de TI e abrange várias indústrias.

Exemplo:Imagine que você é um gerente de marketing analisando dados de comportamento do cliente para uma campanha que se aproxima. Com SQL, você consulta o banco de dados rapidamente para descobrir insights valiosos sobre tendências de compra, ajudando sua equipe a elaborar uma estratégia direcionada. Essa capacidade de utilizar dados de maneira eficaz não apenas aumenta sua atratividade no mercado, mas também enriquece significativamente seu processo de tomada de decisões, tornando você um recurso indispensável em sua organização.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

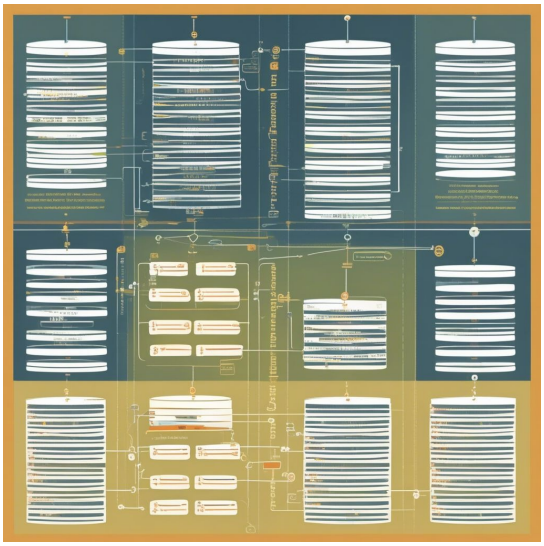
Pensamento crítico

Ponto chave:A Mercado e Utilidade Ampla do SQL

Interpretação crítica:O capítulo enfatiza a crescente demanda por habilidades em SQL no panorama digital em evolução, sugerindo que não apenas profissionais de TI, mas também líderes empresariais, devem aprender SQL para aproveitar os dados em busca de insights estratégicos. No entanto, essa afirmação pode ignorar outras ferramentas analíticas disponíveis atualmente, que podem oferecer vantagens diferentes no processamento de dados e na geração de insights, como Python ou R para análise estatística, conforme discutido em fontes como 'Data Science for Business' de Foster Provost e Tom Fawcett. Os leitores devem avaliar criticamente se focar apenas no SQL é a melhor abordagem para suas trajetórias profissionais específicas.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Capítulo 2 Resumo : 2. Bancos de Dados



Capítulo	Tema	Resumo
2	O que é um Banco de Dados?	Um banco de dados é uma coleção estruturada de dados, referindo-se principalmente a sistemas de gerenciamento de banco de dados relacionais (SGBDR) que organizam dados em tabelas inter-relacionadas.
2	Explorando Bancos de Dados Relacionais	Tabelas têm linhas e colunas e podem se relacionar umas com as outras para recuperação eficiente de dados, semelhante ao uso de PROCV no Excel.
2	Por que Separar Tabelas?	A normalização é fundamental para evitar redundância e dados acumulados, permitindo atualizações mais gerenciáveis ao separar informações relacionadas em tabelas distintas.
2	Escolhendo uma Solução de Banco de Dados	Bancos de dados podem ser leves (como SQLite e Microsoft Access) para pequenas equipes e opções centralizadas (como MySQL, SQL Server e Oracle) para grandes bases de usuários.
2	Bancos de Dados Leves	Bancos de dados leves são fáceis de usar, como o SQLite, adequado para menos usuários, enquanto o Microsoft Access é limitado em escalabilidade se comparado ao SQLite.
2	Bancos de Dados Centralizados	Bancos de dados centralizados são essenciais para aplicações maiores, exigindo hardware de nível servidor e gestão de TI para lidar efetivamente com múltiplos usuários.
2	Nota Importante sobre SQL	SQL é a linguagem universal para interação com dados, consistente entre plataformas, tornando o conhecimento de SQL transferível.
2	Conclusão	Entender as estruturas de banco de dados prepara os leitores para uma interação eficaz com bancos de dados, com foco no SQLite no livro.
3	O que é SQLite?	SQLite é um banco de dados leve adequado para configurações simples, utilizado em vários dispositivos e ideal para protótipos.
3	SQLiteStudio	SQLiteStudio é uma ferramenta intuitiva para gerenciar bancos de dados SQLite, que não requer instalação e permite fácil integração e exploração.
3	Importando e Navegando em Bancos de Dados	SQLiteStudio facilita a adição e navegação de bancos de dados, permitindo que os usuários manipulem dados sem precisar escrever consultas SQL.
3	Entendendo as	Tabelas de banco de dados têm estruturas definidas, onde colunas são atribuídas a tipos de

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Capítulo	Tema	Resumo
	Estruturas de Tabelas	dados específicos, importante para escrever consultas SQL eficazes.

Capítulo 2: Bancos de Dados

O Que É um Banco de Dados?

Um banco de dados é qualquer coleção estruturada de dados, que pode variar de planilhas a arquivos de texto simples, geralmente referindo-se a um sistema de gerenciamento de banco de dados relacional (SGBDR) em contextos profissionais. Um SGBDR organiza os dados em tabelas que podem ser inter-relacionadas.

Explorando Bancos de Dados Relacionais

Tabelas consistem em linhas e colunas, semelhantes a planilhas, e podem se relacionar entre si—for instance, uma tabela de PEDIDOS contendo CUSTOMER_ID que aponta para uma tabela de CLIENTES. Essa relação permite uma recuperação de dados eficiente, semelhante ao VLOOKUP no Excel.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Por Que Tabelas Separadas?

As tabelas são separadas para normalização, evitando redundância; todas as informações armazenadas em uma única tabela podem levar a dados inchados e desafios de manutenção. Por exemplo, separar os dados dos clientes dos pedidos evita a inserção repetitiva de dados. Quando normalizados, apenas um registro por cliente precisa ser atualizado se seus dados mudarem.

Escolhendo uma Solução de Banco de Dados

Os bancos de dados são categorizados em soluções leves e centralizadas. Bancos de dados leves, como SQLite e Microsoft Access, são ideais para usuários individuais ou pequenas equipes, enquanto bancos de dados centralizados são adequados para bases de usuários maiores, lidando com múltiplas solicitações simultâneas de forma eficiente. Bancos de dados centralizados comuns incluem MySQL, Microsoft SQL Server e Oracle.

Bancos de Dados Leves

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Bancos de dados leves são ideais para menos usuários porque são simples, muitas vezes baseados em arquivos e fáceis de compartilhar. O SQLite é destacado por sua acessibilidade e é comumente utilizado em dispositivos. O Microsoft Access também é mencionado por sua usabilidade nos negócios, mas apresenta limitações em escalabilidade em comparação ao SQLite.

Bancos de Dados Centralizados

Para aplicações maiores, bancos de dados centralizados em servidores (ex.: MySQL, Microsoft SQL Server) são necessários. Esses bancos de dados gerenciam múltiplos usuários simultâneos, exigindo hardware de nível servidor e gerenciamento de TI para desempenho ideal.

Nota Importante sobre SQL

SQL serve como a linguagem universal para interagir com dados em todas as plataformas. Diferentes plataformas de banco de dados podem ter suas nuances, mas as consultas SQL permanecem em grande parte consistentes, tornando o conhecimento transferível.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Conclusão

Com uma compreensão das estruturas e soluções de bancos de dados, os leitores podem começar a trabalhar com bancos de dados de forma eficaz. Neste livro, o SQLite será o foco principal para aprender SQL e interação com bancos de dados.

Capítulo 3: SQLite

O Que É SQLite?

SQLite é um banco de dados amplamente distribuído ideal para configurações simples onde baixo overhead e facilidade de uso são desejados. Ele é utilizado em vários dispositivos, incluindo smartphones e aeronaves, e é uma ótima escolha para protótipos.

SQLiteStudio

SQLiteStudio é recomendado para gerenciar bancos de dados SQLite devido à sua interface intuitiva. Não requer instalação

Mais Livros Grátis
Escaneie para baixar
Ouça-o

e integra facilmente vários bancos de dados para exploração e gerenciamento.

Importando e Navegando em Bancos de Dados

Ao iniciar o SQLiteStudio, os usuários podem adicionar bancos de dados, navegar por tabelas e visualizar dados dentro da interface amigável, permitindo tanto a estrutura quanto a manipulação de dados sem SQL.

Compreendendo Estruturas de Tabela

As tabelas de banco de dados têm estruturas definidas, com colunas atribuídas a tipos de dados específicos (ex.: INTEIRO, TEXTO). Familiaridade com essas estruturas é essencial para a redação eficaz de consultas SQL, aprimorando a capacidade do usuário de interagir com os dados de forma eficiente.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Capítulo 3 Resumo : 3. SQLite

Capítulo 3: SQLite

O que é o SQLite?

SQLite é um mecanismo de banco de dados simples, baseado em arquivos, amplamente utilizado em diversos dispositivos, como smartphones, sistemas operacionais e aeronaves, devido à sua simplicidade e baixo consumo de recursos. No entanto, não é ideal para ambientes multiusuário, uma vez que não possui gerenciamento de servidor para acesso simultâneo.

SQLiteStudio

O SQLiteStudio é recomendado como um editor SQL intuitivo para gerenciar bancos de dados SQLite. Não requer instalação e pode ser baixado de seu site oficial. Os usuários devem criar uma pasta para os arquivos do banco de dados, onde podem adicionar e gerenciar bancos de dados facilmente.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Importando e Navegando pelos Bancos de Dados

Ao iniciar o SQLiteStudio, os usuários encontrarão um painel com um navegador de banco de dados e uma área de trabalho SQL. Os bancos de dados podem ser adicionados baixando bancos de dados de exemplo e importando-os para o SQLiteStudio. Os usuários podem explorar o conteúdo dos bancos de dados, acessando tabelas e visualizações, e visualizar informações detalhadas sobre as estruturas das tabelas, incluindo tipos de dados.

Tipos de Dados no SQLite

Compreender os tipos de dados é crucial, especialmente porque cada coluna tem tipos específicos, como INTEGER ou TEXT. Os usuários também podem visualizar e editar dados diretamente no aplicativo.

Instalar o aplicativo Bookey para desbloquear texto completo e áudio

Mais Livros Grátis

Escaneie para baixar

Ouça-o

Ad

Escanear para baixar

11:12
Capítulo 1 de 5
Visão Geral
Olá, bem-vindo ao Bookey. Hoje vamos explorar o livro Comunicação Não Violenta: Uma Linguagem de Vida.
Nós nos comunicamos com os outros todos os dias. Se prestarmos mais atenção às maneiras como falamos e aos nossos sentimentos em relação a várias formas de conversa, perceberemos que algumas palavras podem ser extremamente prejudiciais. Por exemplo, há negações verbais, interrupções rudes e julgamentos casuais. Agora, o livro Comunicação Não Violenta: Uma Linguagem de Vida nos ensinará uma nova maneira de comunicação, permitindo-nos evitar tais danos e obter mais amor e felicidade
11:12
Comunicação Não Violenta
Uma linguagem da vida
Marshall Rosenberg
Descrição
Ler

Por que o Bookey é um aplicativo indispensável para amantes de livros

Conteúdo de 30min
Quanto mais profunda e clara for a interpretação que fornecemos, melhor será sua compreensão de cada título.

Clipes de Ideias de 3min
Impulsione seu progresso.

Questionário
Verifique se você dominou o que acabou de aprender.

E mais
Várias fontes, Caminhos em andamento, Coleções...

Teste gratuito com Bookey

Capítulo 4 Resumo : 4. SELECT

Capítulo	Conteúdo
Capítulo 4: SELECT	Visão Geral do SELECT: Fundamental para recuperar e transformar dados. Recuperando Dados com SQL: Use o SQLiteStudio para acessar bancos de dados e executar consultas SELECT (por exemplo, `SELECT * FROM CUSTOMER;`). Selecionando Colunas Específicas: Especifique colunas a serem recuperadas (por exemplo, `SELECT CUSTOMER_ID, NAME FROM CUSTOMER;`). Expressões em Instruções SELECT: Realize cálculos (por exemplo, calculando TAXED_PRICE). Aliases e Arredondamento: Use AS para aliases e round() para limitar as casas decimais. Operadores Matemáticos: Resumo dos operadores: +, -, *, /, %. Concatenação de Texto: Combine strings usando `||` (por exemplo, `SELECT NAME, CITY || ', ' || STATE AS LOCATION FROM CUSTOMER;`). Resumo: Utilizando SELECT para recuperar, transformar dados, selecionar colunas específicas e usar expressões.
Capítulo 5: WHERE	Introdução ao WHERE: Usado para filtrar registros em consultas SELECT. Filtrando Registros: Acesse o banco de dados `weather_stations` e consulte (por exemplo, `SELECT * FROM station_data;`). Filtragem Numérica: Use `BETWEEN`, `AND`, `OR` (por exemplo, filtre por ano). Filtragem de Texto: Envolva campos de texto entre aspas simples (por exemplo, `SELECT * FROM station_data WHERE report_code = '513A63';`). Valores Booleanos: Use 1 para verdadeiro, 0 para falso ao filtrar. Tratando Valores NULL: Verifique NULLs com `IS NULL` ou `IS NOT NULL`. Combinando Condições: Use parênteses para clareza em condições complexas. Resumo: Ênfase na filtragem de registros, gerenciamento de NULLs e agrupamento de condições.

Capítulo 4: SELECT

Visão Geral do SELECT

A instrução SELECT é fundamental para recuperar dados de

Mais Livros Grátis
Escaneie para baixar
Ouça-o

bancos de dados. Ela faz mais do que apenas exibir dados; também pode transformá-los em resumos significativos.

Recuperando Dados com SQL

Para começar a usar SQL, acesse o SQLiteStudio e abra o banco de dados `rexon_metals`. O ambiente de trabalho possui um Editor SQL para escrever consultas e um painel de Resultados de Consulta para exibir os resultados. Uma instrução SELECT simples pode ser executada, por exemplo, `SELECT * FROM CUSTOMER;`, que recupera todos os registros da tabela CUSTOMER.

Selecionando Colunas Específicas

Você não precisa selecionar todas as colunas. Pode especificar quais colunas deseja recuperar, como `SELECT CUSTOMER_ID, NAME FROM CUSTOMER;`.

Expressões em Instruções SELECT

Você também pode realizar cálculos dentro de uma instrução SELECT. Por exemplo, para calcular um PREÇO_COM_IMPOSTO que seja 7% maior que o PREÇO

Mais Livros Grátis
Escaneie para baixar
Ouça-o

na tabela PRODUTO, você pode escrever:
`SELECT PRODUCT_ID, DESCRIPTION, PRICE, PRICE * 1.07 AS TAXED_PRICE FROM PRODUCT;`

Aliases e Arredondamento

Você pode usar aliases (usando a palavra-chave AS) para dar nomes personalizados a colunas calculadas ou existentes. Além disso, a função `round()` do SQLite pode ser usada para limitar casas decimais para uma apresentação melhor.

Operadores Matemáticos

Um resumo dos operadores matemáticos em SQL inclui:
-
+
 para adição
-
-
 para subtração
-
\
* para multiplicação
-

Mais Livros Grátis

Escaneie para baixar
Ouça-o

/
 para divisão
-
%
 para operação de resto

Concatenação de Texto

O SQL permite que você concatene strings usando o pipe duplo (`||`). Por exemplo:
`SELECT NAME, CITY || ', ' || STATE AS LOCATION FROM CUSTOMER;` combina cidade e estado.

Resumo

Neste capítulo, aprendemos a utilizar a instrução SELECT para recuperar e transformar dados, selecionar colunas específicas e usar expressões para cálculos.

Capítulo 5: WHERE

Introdução ao WHERE

Mais Livros Grátis
Escaneie para baixar
Ouça-o

A instrução WHERE é usada para filtrar registros com base em critérios especificados e pode ser combinada com consultas SELECT para recuperação de dados mais refinada.

Filtrando Registros

Abra o banco de dados `weather_stations` e consulte os dados usando a instrução `SELECT * FROM station_data;`.

Filtragem Numérica

A filtragem por anos específicos, intervalos e o uso de operadores como `BETWEEN`, `AND` e `OR` são abordados. Por exemplo, para recuperar registros do ano de 2010:
`SELECT * FROM station_data WHERE year = 2010;`

Filtragem de Texto

Os campos de texto nas instruções WHERE devem ser envoltos em aspas simples. Para filtrar por `report_code`, você pode escrever:
`SELECT * FROM station_data WHERE report_code =

Mais Livros Grátis
Escaneie para baixar
Ouça-o

'513A63';`

Valores Booleanos

Para condições Booleanas, use 1 para verdadeiro e 0 para falso no SQLite. Consultas podem ser executadas para filtrar esses valores de forma eficaz.

Tratando Valores NULL

Os NULLs devem ser verificados com `IS NULL` ou `IS NOT NULL`, em vez do operador de igualdade. Lidar corretamente com valores NULL é crucial para consultas de dados precisas.

Combinando Condições

Ao encadear várias condições, use parênteses para clareza e correção dos agrupamentos lógicos, especialmente ao combinar instruções AND e OR.

Resumo

Este capítulo enfatizou a capacidade de filtrar registros

Mais Livros Grátis
Escaneie para baixar
Ouça-o

usando cláusulas WHERE, incluindo o uso de intervalos numéricos, filtragem de texto, valores booleanos, tratamento de NULLs e agrupamento de condições complexas. O próximo capítulo foca na agregação de dados para insights mais profundos.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Exemplo

Ponto chave:Filtragem de Dados é Essencial

Exemplo:Ao utilizar a cláusula WHERE, imagine que você está selecionando apenas os registros que precisa, como estudar apenas os dados de temperatura de 2020 em seu banco de dados meteorológico. Você pode refinar suas consultas para obter dados precisos sem sobras desnecessárias.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Pensamento crítico

Ponto chave:A importância da instrução SELECT em SQL

Interpretação crítica:O capítulo destaca a importância de dominar a instrução SELECT para uma recuperação e transformação de dados eficazes. Embora o autor a apresente como fundamental, os leitores devem refletir sobre a natureza em evolução das práticas de gerenciamento de dados, que podem desafiar a ideia de que esse único método é suficiente em todos os cenários. Por exemplo, em bancos de dados distribuídos modernos e armazéns de dados, a utilidade do SELECT pode ser limitada pela necessidade de consultas mais complexas e técnicas de processamento de dados (Forbes, "Cinco desafios chave no gerenciamento de dados" 2022). Assim, é essencial complementar o conhecimento básico com habilidades avançadas para navegar nas complexidades dos ambientes de dados contemporâneos.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Capítulo 5 Resumo : 5. WHERE

Capítulo	Conteúdo
Capítulo 5: WHERE	Visão Geral: Filtrando registros usando a cláusula WHERE para uma navegação significativa no conjunto de dados. Filtrando Registros: Usando o banco de dados "weather_stations" e a tabela STATION_DATA para consultas de dados direcionadas. Usando WHERE em Números: Exemplos para filtrar anos específicos (por exemplo, 2010) e usando BETWEEN. Instruções AND, OR e IN: Combinando operadores lógicos para criar consultas complexas (por exemplo, filtrar por meses). Usando WHERE em Texto: Literais de string com aspas simples e correspondência de padrões com o operador LIKE. Usando WHERE em Booleanos: Consultando campos booleanos e usando NOT para filtrar registros. Tratamento de NULL: Tratamento especial para valores NULL com IS NULL e IS NOT NULL. Condicionais de Agrupamento: Use parênteses para clareza ao combinar múltiplas condições. Resumo: Importância da cobertura da cláusula WHERE para um filtragem eficiente de dados em SQL; introduz o próximo capítulo sobre agregação de dados.
Capítulo 6: GROUP BY e ORDER BY	Visão Geral: Discute a agregação de dados (resumindo dados em SQL) com operações-chave (soma, mínimo, máximo, contagem, média). Agrupando Registros: Contando registros em uma tabela e filtrando contagens com base em critérios. Ordenando Registros: Usando ORDER BY para classificar dados, com exemplos mostrando a agregação em ordem decrescente. Funções Agregadas: Introdução a COUNT, SUM, MIN, MAX, AVG, contando valores não nulos com COUNT. Instrução HAVING: Filtrando valores agregados usando HAVING em vez de WHERE. Obtendo Registros Distintos: Eliminando duplicatas usando a palavra-chave DISTINCT. Resumo: Cobertura da agregação, ordenação e cláusula HAVING para aprimorar as capacidades de consulta, incentivando a aplicação prática de SQL.

Capítulo 5: WHERE

Visão Geral

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Neste capítulo, nos aprofundamos na filtragem de registros no SQL utilizando a cláusula WHERE. A cláusula WHERE nos permite especificar critérios para selecionar registros, tornando a navegação pelo conjunto de dados mais significativa. Várias funções e expressões podem ser usadas dentro desta cláusula, abrindo caminho para consultas de dados mais detalhadas.

Filtrando Registros

Para começar, você trabalhará com um banco de dados chamado "weather_stations" contendo uma tabela chamada STATION_DATA. Executar uma consulta como `SELECT * FROM station_data;` revela aproximadamente 28.000 registros para revisão. Em vez de examinar cada registro individualmente, usar a cláusula WHERE permite consultas direcionadas.

Usando WHERE em Números

A cláusula WHERE pode filtrar dados com base em campos numéricos facilmente:
- Por exemplo, para recuperar registros específicos do ano 2010:

Mais Livros Grátis
Escaneie para baixar
Ouça-o

`SELECT * FROM station_data WHERE year = 2010;`
- Para excluir aquele ano:
`SELECT * FROM station_data WHERE year != 2010;`
ou
`SELECT * FROM station_data WHERE year <> 2010;`
Intervalos inclusivos podem ser filtrados usando o operador BETWEEN, ou combinando operadores AND para mais especificidade.

Declarações AND, OR e IN

Você também pode usar operadores lógicos como AND e OR para criar consultas complexas:
- Para filtrar meses seletivamente, você pode usar:
`SELECT * FROM station_data WHERE MONTH IN (3,6,9,12);`

Usando WHERE em Texto

Operações de texto exigem que literais de string sejam envolvidas entre aspas simples:
- Por exemplo,
`SELECT * FROM station_data WHERE report_code =

Mais Livros Grátis
Escaneie para baixar
Ouça-o

'513A63';`
- As operações de texto também permitem correspondência de padrões de string por meio do operador LIKE:
  - Exemplo: `SELECT * FROM station_data WHERE report_code LIKE 'A%';`

Usando WHERE em Booleanos

Campos booleanos podem ser consultados diretamente, onde verdadeiro é tipicamente representado como 1 e falso como 0. Você também pode utilizar NOT para filtrar registros indesejados:
- Exemplo para filtrar tornados e granizo:
  `SELECT * FROM station_data WHERE tornado AND hail;`

Manipulando NULL

Valores NULL precisam de consideração especial, pois não podem ser comparados usando operadores padrão. Em vez disso, deve-se usar IS NULL ou IS NOT NULL:
- Exemplo:
  `SELECT * FROM station_data WHERE snow_depth IS NULL;`

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Você pode incluir valores NULL em suas consultas usando OR:
- Exemplo:
  `SELECT * FROM station_data WHERE precipitation IS NULL OR precipitation <= 0.5;`

Condições de Agrupamento

Ao combinar múltiplas condições, utilize parênteses para garantir que o SQL as execute corretamente. Por exemplo, ao checar condições para chuva gelada:
```
SELECT * FROM station_data WHERE (rain = 1 AND temperature <= 32) OR snow_depth > 0;
```
Agrupar mantém a clareza e a precisão na execução.

Resumo

Este capítulo enfatiza a importância da cláusula WHERE para uma filtragem eficiente de dados no SQL. Compreender os fundamentos dos operadores lógicos, funções de texto, manipulação de null e agrupamento de condições estabelece uma base sólida para aprendizado adicional em SQL. O

Mais Livros Grátis
Escaneie para baixar
Ouça-o

próximo capítulo explorará a agregação de dados, ampliando ainda mais suas capacidades em SQL.

Capítulo 6: GROUP BY e ORDER BY

Visão Geral

Este capítulo discute a agregação de dados, comumente referida como condensar ou resumir dados no SQL. As principais operações de agregação incluem soma, mínimo, máximo, contagem e média.

Agrupando Registros

Você pode contar registros em uma tabela:
- Exemplo:
  `SELECT COUNT(*) AS record_count FROM station_data;`
- Para filtrar e contar com base em critérios específicos:
  `SELECT COUNT(*) AS record_count FROM station_data WHERE tornado = 1;`

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Ordenando Registros

Uma vez que os dados estão agregados, podem não aparecer em uma ordem natural. A cláusula ORDER BY ajuda a classificar resultados. A ordenação padrão é ascendente (ASC):
- Para classificar dados do ano mais recente:
  ```
  SELECT year, month, COUNT(*) AS record_count
  FROM station_data
  WHERE tornado = 1
  GROUP BY year, month
  ORDER BY year DESC, month;
  ```

Funções de Agregação

Além de COUNT, outras funções de agregação incluem SUM, MIN, MAX e AVG. A função COUNT pode também contar valores não nulos quando usada com uma coluna específica:
- Exemplo:
  `SELECT COUNT(snow_depth) as recorded_snow_depth_count FROM STATION_DATA;`

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Cláusula HAVING

A cláusula HAVING é usada para filtrar valores agregados após o cálculo, ao contrário da cláusula WHERE:

```
SELECT year, SUM(precipitation) as total_precipitation
FROM station_data
GROUP BY year
HAVING total_precipitation > 30;
```

Obtendo Registros Distintos

A palavra-chave DISTINCT ajuda a eliminar duplicatas dos resultados da consulta:

- Exemplo:
  `SELECT DISTINCT station_number FROM station_data;`

Resumo

Neste capítulo, a agregação e ordenação usando GROUP BY e ORDER BY é abordada em profundidade, utilizando

Mais Livros Grátis
Escaneie para baixar
Ouça-o

funções de agregação para resumir dados de maneira eficiente. A introdução da cláusula HAVING permite a filtragem de dados agregados, aprimorando as capacidades de consulta. Os alunos são incentivados a experimentar com SQL para reforçar seu aprendizado e explorar outras funções que estão descritas nos apêndices para aplicação prática.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Exemplo

Ponto chave:Filtrar dados de forma eficiente é crucial para insights significativos.

Exemplo:Imagine que você foi designado para analisar dados meteorológicos e acabou de acessar todo o conjunto de dados que contém milhares de registros. Em vez de vasculhar os detalhes de cada registro, você pode usar a cláusula WHERE para criar uma consulta focada que filtra todos os dados do, digamos, ano de 2010. Ao executar `SELECT * FROM station_data WHERE year = 2010;`, você está imediatamente concentrando-se nas informações específicas de que precisa, tornando muito mais eficiente a extração de insights. Essa capacidade de filtrar registros através de critérios direcionados transforma a exploração de dados de uma tarefa assustadora em uma atividade simplificada, permitindo que você se concentre no que realmente importa.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Pensamento crítico

Ponto chave:A cláusula WHERE é essencial para a recuperação direcionada de dados

Interpretação crítica:Enquanto o Capítulo 5 enfatiza a importância da cláusula WHERE na filtragem eficaz de registros em consultas SQL, deve-se avaliar criticamente a noção de que seu uso é simples. A declaração do SQL de que permite consultas direcionadas pode simplificar demais as complexidades envolvidas na elaboração de consultas eficazes, especialmente com grandes conjuntos de dados. Questões como otimização de desempenho, entendimento do impacto da indexação e as intricacias das operações de junção em SQL podem complicar os esforços de filtragem. O autor apresenta um forte argumento para a utilidade da cláusula WHERE, mas os leitores devem consultar recursos adicionais, como "SQL Performance Explained" de Markus Winand ou "SQL Queries for Mere Mortals" de John Viescas, para apreciar plenamente a profundidade dos desafios enfrentados na formulação de consultas SQL. Assim, enquanto o conhecimento fundamental é bem apresentado, os profissionais são incentivados a

Mais Livros Grátis
Escaneie para baixar
Ouça-o

aprofundar sua compreensão por meio de recursos variados.

Capítulo 6 Resumo : 6. GROUP BY e ORDER BY

Capítulo 6: GROUP BY e ORDER BY

Agregando Dados

Agregação de dados envolve criar totais a partir dos registros, utilizando operações como SOMA, MÍNIMO, MÁXIMO, CONTAR e MÉDIA. O SQL permite agrupar esses totais com base em colunas especificadas para uma análise de dados eficaz.

Agrupando Registros

Para contar registros em uma tabela, execute:
```sql
SELECT COUNT(*) AS contagem_registros FROM station_data;
```
Utilize condições com COUNT() para filtrar dados

Mais Livros Grátis
Escaneie para baixar
Ouça-o

específicos, como contar ocorrências de tornados:

```sql
SELECT COUNT(*) AS contagem_registros FROM station_data WHERE tornado = 1;
```

Para categorizar as contagens por ano, use:

```sql
SELECT ano, COUNT(*) AS contagem_registros FROM station_data WHERE tornado = 1 GROUP BY ano;
```

Para contagens por ano e mês, ajuste a consulta:

```sql
SELECT ano, mes, COUNT(*) AS contagem_registros FROM station_data WHERE tornado = 1 GROUP BY ano, mes;
```

Ordenando Registros

Instalar o aplicativo Bookey para desbloquear texto completo e áudio

Mais Livros Grátis

Escaneie para baixar

Ouça-o

Ad

Escanear para baixar

App Store
Escolha dos Editores

22k avaliações de 5 estrelas

Feedback Positivo

Afonso Silva

s cada resumo de livro não só
o, mas também tornam o
m divertido e envolvente. O
tou a leitura para mim.

Fantástico! Brígida Santos

Estou maravilhado com a variedade de livros e idiomas que o Bookey suporta. Não é apenas um aplicativo, é um portal para o conhecimento global. Além disso, ganhar pontos para caridade é um grande bônus!

Fi
O
só
o
O

na Oliveira

correr as
ém me dá
omprar a
ar!

Adoro! Duarte Costa

Usar o Bookey ajudou-me a cultivar um hábito de leitura sem sobrecarregar minha agenda. O design do aplicativo e suas funcionalidades são amigáveis, tornando o crescimento intelectual acessível a todos.

Economiza tempo!

O Bookey é o meu apli
crescimento intelectua
perspicazes e lindame
um mundo de conheci

Aplicativo incrível! Estevão Pereira

Eu amo audiolivros, mas nem sempre tenho tempo para ouvir o livro inteiro! O Bookey permite-me obter um resumo dos destaques do livro que me interessa!!! Que ótimo conceito!!! Altamente recomendado!

Aplicativo lindo

Este aplicativo é um salva-vidas para
de livros com agendas lotadas. Os re
precisos, e os mapas mentais ajudar
o que aprendi. Altamente recomend

Teste gratuito com Bookey

Capítulo 7 Resumo : 7. Instruções CASE

Capítulo 7. Instruções CASE

Introdução às Instruções CASE

Antes de mergulhar nos operadores JOIN, é essencial entender a instrução CASE, que permite substituições de valores com base em condições.

A Instrução CASE

Uma instrução CASE permite mapear múltiplas condições a valores correspondentes. Ela começa com "CASE" e termina com "END," especificando condições como "WHEN [condição] THEN [valor]." Um "ELSE" opcional pode ser utilizado para valores padrão se nenhuma das condições corresponder.

Exemplo de Uso

Para categorizar a velocidade do vento em níveis de

Mais Livros Grátis
Escaneie para baixar
Ouça-o

severidade:

```sql
SELECT report_code, year, month, day, wind_speed,
CASE
    WHEN wind_speed >= 40 THEN 'ALTO'
    WHEN wind_speed >= 30 THEN 'MODERADO'
    ELSE 'BAIXO'
END AS wind_severity
FROM station_data;
```

Essa instrução pode ser simplificada omitindo a segunda condição, uma vez que a consulta é avaliada de cima para baixo.

Agregando Instruções CASE

Agrupar instruções CASE leva a transformações de dados poderosas. Por exemplo, agrupar por ano e severidade do vento permite contagens agregadas:

```sql
SELECT year,
    CASE
        WHEN wind_speed >= 40 THEN 'ALTO'
        WHEN wind_speed >= 30 THEN 'MODERADO'
```

Mais Livros Grátis
Escaneie para baixar
Ouça-o

```
        ELSE 'BAIXO'
    END AS wind_severity,
    COUNT(*) AS record_count
FROM station_data
GROUP BY 1, 2;
```

O Truque “Zero/Nulo” da Instrução CASE

Essa técnica aplica filtros diferentes para agregações em uma única consulta. Por exemplo, somar a precipitação para condições de tornado e não tornado pode ser realizado através de:

```sql
SELECT year, month,
    SUM(CASE WHEN tornado = 1 THEN precipitation
ELSE 0 END) AS tornado_precipitation,
    SUM(CASE WHEN tornado = 0 THEN precipitation
ELSE 0 END) AS non_tornado_precipitation
FROM station_data
GROUP BY year, month;
```

Usar nulo em vez de zero pode ajudar em operações de MAX ou MIN para cálculos precisos.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Usando Expressões Booleanas em Instruções CASE

Você pode combinar vários operadores lógicos usando CASE. Aqui está um exemplo para calcular a média das temperaturas com base nas condições de chuva/granizo após 2000:

```sql
SELECT month,
    AVG(CASE WHEN rain OR hail THEN temperature
ELSE NULL END) AS avg_precipitation_temp,
    AVG(CASE WHEN NOT (rain OR hail) THEN
temperature ELSE NULL END) AS
avg_non_precipitation_temp
FROM STATION_DATA
WHERE year > 2000
GROUP BY month;
```

Resumo das Instruções CASE

Este capítulo enfatizou as instruções CASE, destacando sua flexibilidade em trocar valores e melhorar a análise de dados através da agregação. As instruções CASE são fundamentais

Mais Livros Grátis
Escaneie para baixar
Ouça-o

para consultas SQL eficazes, preparando o terreno para a introdução às operações JOIN.

Capítulo 8. JOIN - Unindo Tabelas

Introdução aos JOIN

Unir tabelas é crucial em SQL, pois consolida dados de tabelas distintas em uma visão abrangente, especialmente em bancos de dados normalizados.

Entendendo Relacionamentos Entre Tabelas

As tabelas frequentemente têm relacionamentos de pai e filho. Por exemplo, a tabela CUSTOMER_ORDER possui um CUSTOMER_ID que faz referência à tabela CUSTOMER.

INNER JOIN

O INNER JOIN mescla registros onde há uma correspondência em ambas as tabelas. Por exemplo:

Mais Livros Grátis
Escaneie para baixar
Ouça-o

```sql
SELECT ORDER_ID, CUSTOMER.CUSTOMER_ID,
ORDER_DATE, SHIP_DATE, NAME,
STREET_ADDRESS,
CITY, STATE, ZIP, PRODUCT_ID, ORDER_QTY
FROM CUSTOMER
INNER JOIN CUSTOMER_ORDER ON
CUSTOMER.CUSTOMER_ID =
CUSTOMER_ORDER.CUSTOMER_ID;
```

Isso inclui apenas registros que existem em ambas as tabelas.

LEFT JOIN

Em cenários onde você deseja incluir todos os registros de uma tabela, independentemente de haver uma correspondência na outra, usa-se um LEFT JOIN:

```sql
SELECT CUSTOMER.CUSTOMER_ID, NAME AS
CUSTOMER_NAME
FROM CUSTOMER
LEFT JOIN CUSTOMER_ORDER ON
CUSTOMER.CUSTOMER_ID =
CUSTOMER_ORDER.CUSTOMER_ID
```

Mais Livros Grátis
Escaneie para baixar
Ouça-o

WHERE ORDER_ID IS NULL;
```

Essa consulta retorna clientes mesmo que não tenham pedidos associados, com campos nulos para dados ausentes.

**Outros Tipos de JOIN**

-
**RIGHT JOIN**
: Inclui todos os registros da tabela da direita.
-
**FULL OUTER JOIN**
: Combina LEFT e RIGHT JOIN, mas é menos comum.

**Unindo Múltiplas Tabelas**

Relações complexas muitas vezes requerem múltiplas junções. POR EXEMPLO, unir as tabelas CUSTOMER, CUSTOMER_ORDER e PRODUCT permite uma recuperação abrangente de dados:
```sql
SELECT ORDER_ID, CUSTOMER.CUSTOMER_ID,
NAME AS CUSTOMER_NAME, ORDER_QTY,
ORDER_QTY * PRICE AS REVENUE

Mais Livros Grátis
Escaneie para baixar
Ouça-o

```
FROM CUSTOMER
INNER JOIN CUSTOMER_ORDER ON
CUSTOMER.CUSTOMER_ID =
CUSTOMER_ORDER.CUSTOMER_ID
INNER JOIN PRODUCT ON
CUSTOMER_ORDER.PRODUCT_ID =
PRODUCT.PRODUCT_ID;
```

Agregando JOINs

Para agregar dados por cliente enquanto une múltiplas tabelas:

```sql
SELECT CUSTOMER.CUSTOMER_ID, NAME AS
CUSTOMER_NAME, COALESCE(SUM(ORDER_QTY *
PRICE), 0) AS TOTAL_REVENUE
FROM CUSTOMER
LEFT JOIN CUSTOMER_ORDER ON
CUSTOMER.CUSTOMER_ID =
CUSTOMER_ORDER.CUSTOMER_ID
LEFT JOIN PRODUCT ON
CUSTOMER_ORDER.PRODUCT_ID =
PRODUCT.PRODUCT_ID
```

Mais Livros Grátis
Escaneie para baixar
Ouça-o

```
GROUP BY 1, 2;
```

Essa abordagem garante que todos os clientes sejam exibidos com sua receita total, padrão em zero se nenhum existir.

Resumo dos Joins

As operações JOIN são fundamentais em SQL, tornando possível unir diversas tabelas em um único e informativo conjunto de dados. Compreender e praticar essas operações irá aprimorar consideravelmente suas habilidades em SQL.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Capítulo 8 Resumo : 8. JOIN

Resumo do Capítulo 8: JOIN - Unindo Tabelas

Introdução aos JOINs

- O JOIN é um recurso essencial do SQL que o diferencia de outras tecnologias de dados.
- Compreender bancos de dados relacionais é fundamental, especialmente como as tabelas se relacionam por meio de campos como CUSTOMER_ID.

Relacionamentos entre Tabelas

- O relacionamento entre CUSTOMER e CUSTOMER_ORDER é do tipo pai-filho, onde CUSTOMER fornece informações críticas para CUSTOMER_ORDER.
- O relacionamento mais comum é um-para-muitos, exemplificado por um único cliente tendo múltiplos pedidos.

INNER JOIN

Mais Livros Grátis
Escaneie para baixar
Ouça-o

- A operação INNER JOIN combina duas tabelas com base em um campo comum, tipicamente usando chaves como CUSTOMER_ID.
- Exemplo de consulta SQL:

```sql
SELECT ORDER_ID, CUSTOMER.CUSTOMER_ID, ORDER_DATE, ...
FROM CUSTOMER
INNER JOIN CUSTOMER_ORDER ON CUSTOMER.CUSTOMER_ID = CUSTOMER_ORDER.CUSTOMER_ID
```

- Apenas os registros com chaves correspondentes em ambas as tabelas são retornados, significando que clientes sem pedidos não serão incluídos.

LEFT JOIN

- O LEFT JOIN mantém todos os registros da tabela à esquerda (por exemplo, CUSTOMER), mesmo que não haja correspondência na tabela à direita (por exemplo, CUSTOMER_ORDER).
- Isso ajuda a incluir clientes sem pedidos:

Mais Livros Grátis
Escaneie para baixar
Ouça-o

```sql
SELECT ...
FROM CUSTOMER
LEFT JOIN CUSTOMER_ORDER ON
CUSTOMER.CUSTOMER_ID =
CUSTOMER_ORDER.CUSTOMER_ID
```

Outros Tipos de JOIN

- RIGHT JOIN e FULL OUTER JOIN são menos comuns, mas podem fornecer todos os registros de qualquer lado do JOIN.
- O RIGHT JOIN é semelhante ao LEFT JOIN, mas foca na tabela à direita.

Unindo Múltiplas Tabelas

- Os JOINs podem envolver mais de duas tabelas.
- Por exemplo, a combinação das tabelas CUSTOMER, CUSTOMER_ORDER e PRODUCT permite extrair dados essenciais do dia a dia.

Grupo de JOINs

Mais Livros Grátis
Escaneie para baixar
Ouça-o

- A agregação de dados, como a receita total por cliente, requer agrupamento por identificadores de cliente enquanto mantém os JOINs para contexto adicional.
- `LEFT JOIN` pode ajudar a incluir clientes sem pedidos, evitando mal-entendidos na contabilidade da receita.

Conclusão

- Os JOINs são complexos, mas vitais no SQL, permitindo que os usuários criem visualizações significativas de dados ao unir várias tabelas.
- Dominar os JOINs aprimora a análise de dados e atende aos requisitos de negócios de forma eficaz.

Resumo do Capítulo 9: Planejamento de Design de Banco de Dados

Visão Geral do Design de Banco de Dados

- A transição de operações básicas de SELECT para a criação e gerenciamento de bancos de dados requer um planejamento

Mais Livros Grátis
Escaneie para baixar
Ouça-o

cuidadoso.
- As principais questões de design envolvem identificar os requisitos de negócios, estruturas de tabelas, normalização e relacionamentos.

Povoando Dados

- Considere o volume de dados e como eles serão mantidos, incluindo potencial automação para a entrada de dados.

Considerações de Segurança

- A segurança é crucial, especialmente se o banco de dados será acessível pela web. Os ataques de SQL injection são uma preocupação significativa.
- Controle de acesso, planos de backup de dados e registro de mudanças são componentes vitais da segurança do banco de dados.

Estrutura de Design de Banco de Dados: A Conferência SurgeTech

- Considere as entidades envolvidas: ATTENDEE, COMPANY, PRESENTATION, ROOM e

Mais Livros Grátis
Escaneie para baixar
Ouça-o

PRESENTATION_ATTENDANCE.
- Cada entidade tem atributos associados que informam a estrutura das tabelas.

Chaves Primárias e Estrangeiras

- Designar chaves primárias para tabelas para garantir registros únicos e estabelecer relacionamentos por meio de chaves estrangeiras nas tabelas filhas.

Visualização de Esquema

- Um esquema de banco de dados representa visualmente os relacionamentos e estruturas das tabelas, ajudando a entender como diferentes componentes de dados interagem.

Criando e Gerenciando Tabelas

- Utilize ferramentas como SQLiteStudio para criar tabelas de forma eficiente, garantindo configurações e restrições adequadas.

Definindo Chaves Estrangeiras

Mais Livros Grátis
Escaneie para baixar
Ouça-o

- As chaves estrangeiras mantêm a integridade referencial, garantindo que cada registro filho tenha um registro pai correspondente, evitando registros órfãos.

Criando Views

- Armazenar consultas SELECT usadas com frequência como views simplifica o acesso e reutilização dos dados sem necessariamente modificar a estrutura subjacente dos dados.

Eficiência do Design

- Um banco de dados bem projetado adota normalização e restrições que melhoram a integridade e usabilidade dos dados, estabelecendo uma base sólida para a gestão futura dos dados em capítulos subsequentes.

Este resumo captura os conceitos fundamentais e instruções delineadas nos Capítulos 8 e 9 de "Introduçao a Linguagem SQL", focando em operações JOIN e nos princípios de design eficaz de banco de dados.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Pensamento crítico

Ponto chave:Os JOINs são essenciais para o gerenciamento de bancos de dados relacionais em SQL.

Interpretação crítica:O autor enfatiza a necessidade de dominar as operações de JOIN para trabalhar efetivamente com bancos de dados relacionais, embora se possa argumentar que as complexidades dos JOINs podem levar a problemas de desempenho e mal-entendidos na análise de dados. Por exemplo, enquanto os INNER JOINs filtram registros não correspondentes, isso pode criar conjuntos de dados incompletos que podem induzir os analistas ao erro. A literatura sobre desempenho de banco de dados sugere que, embora os JOINs sejam poderosos, devem ser utilizados com cautela, conforme discutido na pesquisa de Thomas Connolly & Carolyn Begg em "Database Systems: A Practical Approach to Design, Implementation, and Management" (2015). Os leitores são encorajados a avaliar criticamente essas metodologias e pesar suas desvantagens em cenários práticos.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Capítulo 9 Resumo : 9. Design de Banco de Dados

Capítulo 9: Design de Banco de Dados

Planejando um Banco de Dados

Nos capítulos anteriores, o foco foi na leitura de dados com a instrução SELECT sem alterá-los. Agora, vamos nos aprofundar nas operações de escrita, onde iremos criar, inserir, atualizar e deletar dados em nosso banco de dados. Um design bem pensado é crucial para evitar arrependimentos mais tarde. As principais considerações incluem:

-

Perguntas de Design
: Quais são as necessidades do negócio? Quais tabelas são necessárias? Quais colunas essas tabelas terão? Como as tabelas devem ser normalizadas? Qual é a natureza das relações pai/filho?

-

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Perguntas de Dados
: Quanto dado as tabelas manterão? Quem irá preencher os dados? De onde os dados virão? Existem automatizações para preencher as tabelas?
-
Perguntas de Segurança
: Quem acessa o banco de dados? Que permissões os usuários têm? Quais são os planos de backup existentes?

Exemplo de Banco de Dados: A Conferência SurgeTech

O objetivo é projetar um banco de dados para gerenciar participantes, empresas, apresentações, salas e frequência.
1.
PARTICIPANTE
: Rastrear o ID, nome, número de telefone, email e status VIP de cada participante.

Instalar o aplicativo Bookey para desbloquear texto completo e áudio

Mais Livros Grátis

Escaneie para baixar

Ouça-o

Ad

Escanear para baixar

Ler, Compartilhar, Empoderar

Conclua Seu Desafio de Leitura, Doe Livros para Crianças Africanas.

O Conceito

BOOKS FOR AFRICA × ×

Esta atividade de doação de livros está sendo realizada em conjunto com a Books For Africa.Lançamos este projeto porque compartilhamos a mesma crença que a BFA: Para muitas crianças na África, o presente de livros é verdadeiramente um presente de esperança.

A Regra

Ganhe 100 pontos → **Resgate um livro** → **Doe para a África**

Seu aprendizado não traz apenas conhecimento, mas também permite que você ganhe pontos para causas beneficentes! Para cada 100 pontos ganhos, um livro será doado para a África.

Teste gratuito com Bookey

Capítulo 10 Resumo : 10. Gerenciando Dados

Capítulo 10: Gerenciando Dados

Neste capítulo, exploramos como gerenciar efetivamente dados dentro de um banco de dados por meio de três operações principais em SQL: INSERT, DELETE e UPDATE. Um gerenciamento de dados eficaz depende de um bom design de banco de dados, incluindo estrutura de tabelas, restrições de coluna e relacionamentos que asseguram a integridade e a eficiência dos dados.

Operações INSERT

- A instrução INSERT é usada para adicionar novos registros a uma tabela de banco de dados. Você pode escolher quais campos preencher, deixando outros com valores padrão ou como NULL.
- Exemplo: `INSERT INTO ATTENDEE (FIRST_NAME, LAST_NAME) VALUES ('Thomas','Nield')`
- Registros podem ser inseridos em massa separando os

Mais Livros Grátis
Escaneie para baixar
Ouça-o

valores com vírgulas. Isso é mais eficiente do que inserir um registro de cada vez.
- Você também pode inserir dados usando resultados de uma instrução SELECT, desde que os campos correspondam.

Testando Chaves Estrangeiras

- Restrições de chave estrangeira têm um papel vital na manutenção da integridade dos dados. Por exemplo, tentar inserir um registro de EMPRESA com um PRIMARY_CONTACT_ID que não existe deve gerar um erro, confirmando que a restrição de chave estrangeira está funcionando corretamente.

Operações DELETE

- A instrução DELETE remove registros de uma tabela e pode deletar todos os registros ou aqueles que atendem a condições específicas.
- Exemplo de exclusão condicional: `DELETE FROM ATTENDEE WHERE PHONE IS NULL AND EMAIL IS NULL`
- Recomenda-se executar uma instrução SELECT primeiro para visualizar quais registros serão deletados.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

TRUNCATE TABLE

- Embora não suportado no SQLite, TRUNCATE TABLE é uma maneira eficiente em outros sistemas SQL de remover todos os registros e redefinir campos de auto-incremento.

Operações UPDATE

- A instrução UPDATE modifica registros existentes. Você pode atualizar campos individualmente ou em conjunto.
- Exemplo de atualização de múltiplos campos: `UPDATE ATTENDEE SET FIRST_NAME = UPPER(FIRST_NAME), LAST_NAME = UPPER(LAST_NAME)`
- Use condições WHERE para especificar quais registros devem ser atualizados.

DROP TABLE

- O comando DROP TABLE exclui permanentemente uma tabela do banco de dados, tornando-se uma operação poderosa e potencialmente perigosa.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Resumo

Com um entendimento das operações INSERT, DELETE, UPDATE e outras operações em SQL, você está equipado para criar e gerenciar seu próprio banco de dados de forma eficaz. O capítulo final abordará técnicas de gerenciamento de dados eficientes e interfaces de usuário para manipulação de dados.

Mais Livros Grátis

Escaneie para baixar

Ouça-o

Capítulo 11 Resumo : 11. Indo em Frente

Capítulo 11: Indo em Frente

Parabéns por chegar a este capítulo! Você adquiriu um conjunto de habilidades valiosas em SQL e design de bancos de dados, que são aplicáveis em negócios, TI e engenharia. Seu nível de ambição em relação à tecnologia pode variar; queira você ver o SQL como essencial para sua carreira ou como uma habilidade complementar, é fundamental definir suas metas e prioridades profissionais pessoais.

- Se o SQL ressoou significativamente com você, considere explorar aspectos mais profundos, como inteligência de negócios ou desenvolvimento de software e interfaces gráficas de usuário. Existem inúmeros recursos disponíveis para aprendizado adicional, incluindo funcionalidades avançadas de SQL e plataformas de banco de dados como MySQL ou SQLite.

- A Internet é uma rica fonte de tutoriais, guias e assistência comunitária. Aprender tecnologias adicionais, como Python para manipulação de dados ou R para inteligência de negócios, pode aumentar sua adaptabilidade e atratividade no mercado. Python é amigável e eficaz para várias aplicações,

Mais Livros Grátis
Escaneie para baixar
Ouça-o

enquanto R é poderoso para análise estatística e aprendizado de máquina.
- Para aqueles que desejam se aventurar no desenvolvimento completo de software, linguagens como Java, Swift ou C# são recomendadas para a criação de aplicações comerciais. Recursos abrangentes estão disponíveis online, e plataformas como Pluralsight e W3Schools oferecem treinamento fundamental. Google e Stack Overflow também são inestimáveis para solucionar problemas e receber conselhos de profissionais experientes.
- Além disso, feedback é bem-vindo. Você pode entrar em contato por e-mail para perguntas ou comentários sobre o material apresentado neste livro.

Apêndice A: Operadores e Funções

Este apêndice descreve operações e funções chave no SQLite que podem servir como uma referência conveniente.

Apêndice A1: Consultas de Expressões Literais

Você pode testar operadores e funções diretamente utilizando expressões. Por exemplo:
- `SELECT 5 + 7` resulta em 12.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

- `SELECT INSTR('TONY STARK', 'TONY')` verifica a presença de substring.

Apêndice A2: Operadores Matemáticos

Os operadores matemáticos básicos no SQLite incluem:
- Adição (+), Subtração (-), Multiplicação (*), Divisão (/), e Módulo (%).

Apêndice A3: Operadores de Comparação

Os operadores de comparação resultam em verdadeiro (1) ou falso (0):
- Exemplos incluem igualdade (=), desigualdade (!=), maior que (>), menor que (<), etc.

Apêndice A4: Operadores Lógicos

Os operadores lógicos permitem combinar expressões booleanas:
- AND, OR, BETWEEN, IN, NOT, IS NULL, e IS NOT NULL.

Apêndice A5: Operadores de Texto

Mais Livros Grátis
Escaneie para baixar
Ouça-o

O processamento de texto inclui:
- Concatenação (||), LIKE para correspondência de padrões, e REGEXP para correspondência de padrões de texto.

Apêndice A6: Funções Comuns

As funções chave incluem:
- `abs()`, `coalesce()`, `instr()`, `length()`, `trim()`, `random()`, `replace()`, `substr()`, `lower()`, e `upper()`.

Apêndice A7: Funções de Agregação

As funções para agregar dados incluem:
- `avg()`, `count()`, `max()`, `min()`, `sum()`, `group_concat()`.

Apêndice A8: Funções de Data e Hora

Trabalhar com datas e horas no SQLite geralmente envolve:
- Armazenar datas no formato AAAA-MM-DD e horas como HH:MM:SS para comparações e operações precisas. As funções incluem `DATE()`, `TIME()`, e `DATETIME()` para várias manipulações. Documentação completa está disponível para funcionalidades mais detalhadas.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Capítulo 12 Resumo : A. Operadores e Funções

Apêndice A. Operadores e Funções

Este apêndice serve como um guia de referência para operações e funções comuns no SQLite, focando em conceitos universais de SQL. Detalhes abrangentes sobre os recursos do SQLite estão disponíveis no site da documentação do SQLite.

Apêndice A1 – Consultas de Expressões Literais

Operadores e funções podem ser testados sem consultar tabelas, selecionando expressões escalares. Por exemplo, `SELECT 5 + 7` resulta em 12, e `SELECT INSTR('TONY STARK', 'TONY')` verifica a presença de 'TONY'.

Apêndice A2 – Operadores Matemáticos

O SQLite inclui operadores matemáticos básicos:

-

Mais Livros Grátis
Escaneie para baixar
Ouça-o

+
: Adição (por exemplo, `x + y = 10`)
-
-
: Subtração (por exemplo, `x - y = 4`)
-
*
: Multiplicação (por exemplo, `x * y = 21`)
-
/
: Divisão (por exemplo, `x / y = 2`)
-
%
: Resto (por exemplo, `x % y = 1`)

Apêndice A3 – Operadores de Comparação

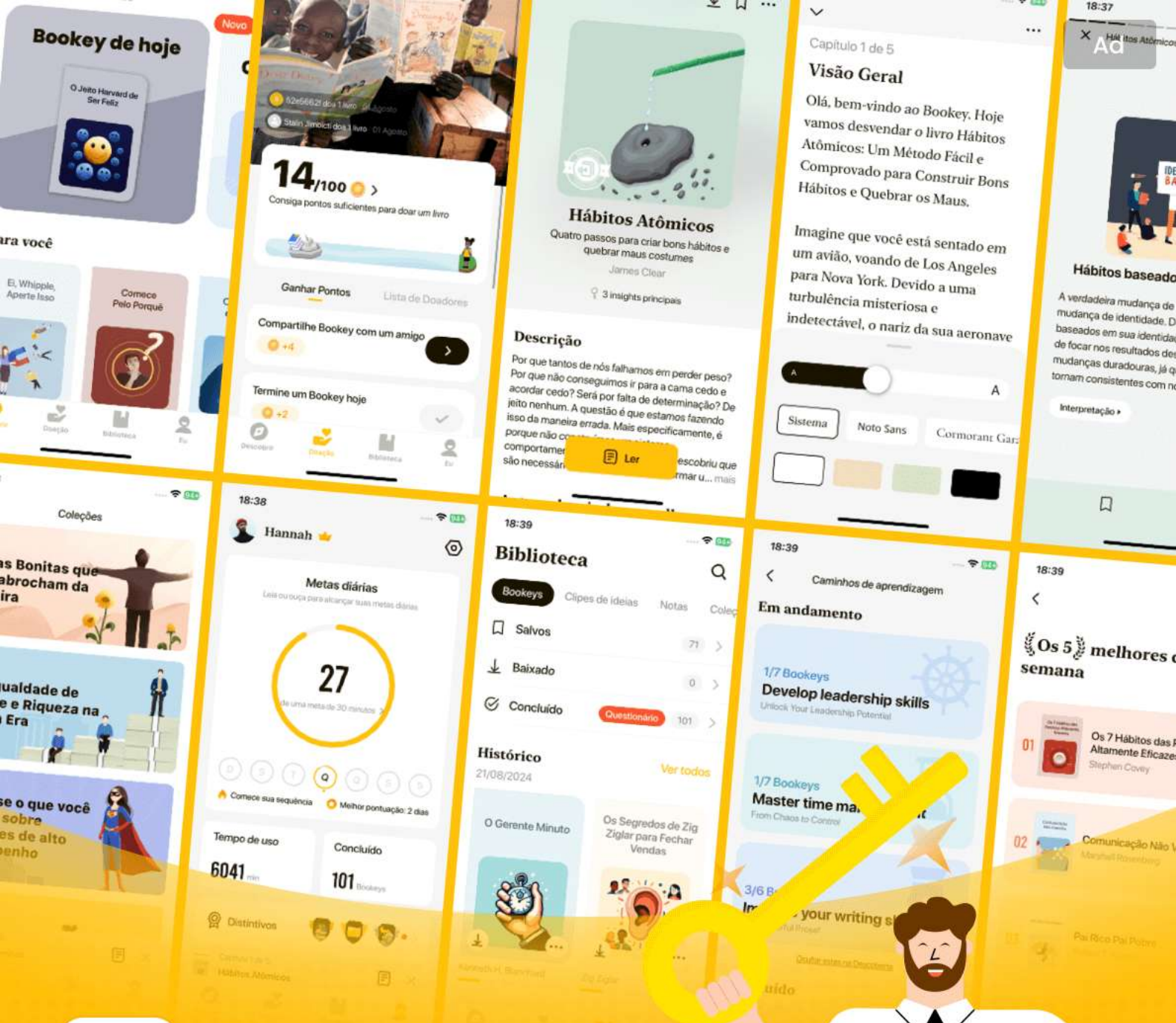
Esses retornam verdadeiro (1) ou falso (0):

Instalar o aplicativo Bookey para desbloquear texto completo e áudio

Mais Livros Grátis

Escaneie para baixar

Ouça-o



As melhores ideias do mundo desbloqueiam seu potencial

Essai gratuit avec Bookey

Descarregar na App Store

OBTER NO Google Play

Escanear para baixar

Capítulo 13 Resumo : B. Tópicos Suplementares

Apêndice B: Tópicos Suplementares

Este apêndice abrange tópicos adicionais de SQL e banco de dados que estão fora do foco principal do livro, visando ajudar os leitores a aprimorar seu conhecimento em SQL.

Apêndice B1: Tópicos de Interesse Adicional

Este guia introdutório de SQL apresenta tópicos fundamentais, mas aqueles que desejam uma exploração mais profunda podem investigar os seguintes tópicos:

-
UNION e UNION ALL
: Combine resultados de várias consultas.

-
Subconsultas
: Execute consultas dentro de consultas como se fossem tabelas.

-

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Índices
: Melhore o desempenho das consultas SELECT para grandes conjuntos de dados (discutido mais adiante no Apêndice B2).
-
Transações
: Agrupe várias instruções INSERT/UPDATE/DELETE em um único lote confiável (mais detalhes no Apêndice B3).
-
Gatilhos
: Automatize respostas a certos eventos de manipulação de dados.
-
Expressões regulares
: Utilize correspondência avançada de padrões de texto.
-
Administração de banco de dados
: Otimize bancos de dados em larga escala em ambientes corporativos.

Apêndice B2: Melhorando o Desempenho com Índices

À medida que os bancos de dados se expandem, o desempenho das consultas SELECT pode diminuir devido ao

Mais Livros Grátis
Escaneie para baixar
Ouça-o

volume de registros processados. Os índices podem aprimorar significativamente as velocidades de busca, semelhante a um índice em um livro, enquanto podem potencialmente reduzir a velocidade de instruções INSERT, UPDATE e DELETE. Essas compensações exigem consideração cuidadosa. Os índices devem ser implementados após o design, quando problemas de desempenho surgirem, concentrando-se em colunas frequentemente consultadas. Por exemplo, para melhorar as buscas em uma coluna de preços, pode-se usar:

```sql
CREATE INDEX price_index ON PRODUCT(price);
```

Os índices podem ser ÚNICOS ou compostos para otimizar consultas que envolvem várias colunas. No entanto, mantenha um equilíbrio: eles são mais eficazes para tabelas grandes e indesejáveis para tabelas pequenas, muito atualizadas.

Apêndice B3: Transações

As transações permitem a execução de múltiplas instruções SQL (INSERT/UPDATE/DELETE) como uma única operação atômica—ou seja, ou têm sucesso totalmente ou

Mais Livros Grátis
Escaneie para baixar
Ouça-o

reverte para preservar a integridade dos dados. Isso é crucial em cenários como transações financeiras para evitar perda de dados. O processo envolve:

1. Iniciando uma transação:
   ```sql
   BEGIN TRANSACTION;
   ```
2. Executando os comandos SQL desejados, que serão registrados.
3. Finalizando as alterações com:
   ```sql
   COMMIT;
   ```
   Se algum comando falhar, use:
   ```sql
   ROLLBACK;
   ```

Esse método protege os dados durante operações complexas ou trabalhos automatizados, garantindo recuperação de erros.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Ad

Escanear para baixar

Experimente o aplicativo Bookey para ler mais de 1000 resumos dos melhores livros do mundo

Desbloqueie 1000+ títulos, 80+ tópicos

Novos títulos adicionados toda semana

duct & Brand | Liderança & Colaboração | Gerenciamento de Tempo | Relacionamento & Comunica

Estratégia de Negócios | Criatividade | Memórias | Conheça a Si Mesmo | Psicologia Po

mpreendedorismo | História Mundial | Comunicação entre Pais e Filhos | Autocuidado | Mi

Visões dos melhores livros do mundo

Teste gratuito com Bookey

Melhores frases do Introduçao a Linguagem SQL por Thomas Nield com números de página

Ver no site do Bookey e gerar imagens de citações bonitas

Capítulo 1 | Frases das páginas 102-128

1. ‘SQL fornece um meio para acessar e manipular esses dados de maneiras significativas e fornecer insights de negócios que antes não eram possíveis.’
2. ‘Nos últimos anos, os dados se tornaram subitamente onipresentes—no entanto, poucas pessoas sabem como acessá-los de forma significativa, o que colocou as habilidades em SQL em alta demanda.’
3. ‘Uma falsa percepção sobre o SQL é que é uma habilidade de TI e, portanto, aplicável apenas a profissionais de tecnologia (não de negócios).’
4. ‘O SQL pode abrir muitos caminhos de carreira porque permite que os indivíduos conheçam melhor seus negócios através dos dados que os impulsionam.’

Capítulo 2 | Frases das páginas 129-200

Mais Livros Grátis

Escaneie para baixar

Ouça-o

1. Na definição mais ampla, um banco de dados é qualquer coisa que coleta e organiza dados.
2. A motivação é a normalização, que consiste em separar os diferentes tipos de dados em suas próprias tabelas, em vez de colocá-los em uma única tabela.
3. Bancos de dados relacionais e SQL não são proprietários.
4. Se você espera que dezenas, centenas ou milhares de usuários e aplicações usem um banco de dados simultaneamente, bancos de dados leves não vão dar conta.
5. SQL é a linguagem universal para trabalhar com dados em todas essas plataformas.

Capítulo 3 | Frases das páginas 201-252

1. SQLite é o banco de dados mais amplamente distribuído no mundo.
2. Ele se destaca onde simplicidade e baixo consumo de recursos são necessários.
3. Dedique um tempo e familiarize-se com o SQLiteStudio.
4. O comando SELECT pode fazer muito mais do que simplesmente selecionar colunas.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

5.Podemos usar aliases para dar nomes a expressões.

6.Esta coluna não é armazenada na tabela, mas sim calculada e exibida para nós sempre que executamos esta consulta.

7.A concatenação deve funcionar com qualquer tipo de dado (números, datas, etc.) e tratá-los como texto ao mesclar.

Mais Livros Grátis

Escaneie para baixar

Ouça-o

Escanear para baixar

Baixe o app Bookey para desfrutar

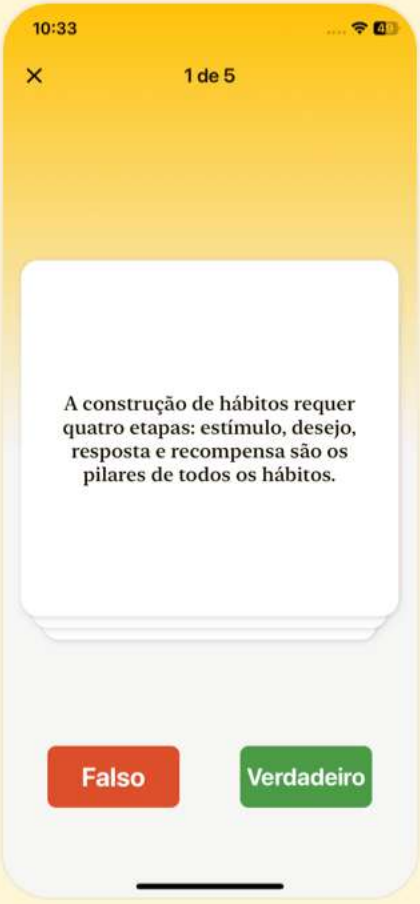
Mais de 1000 resumos de livros com quizzes

Teste grátis disponível!

Descarregar na App Store

OBTER NO Google Play

18:37
Hábitos Atômicos
Quatro passos para criar bons hábitos e quebrar maus costumes
James Clear
3 insights principais
Descrição
Por que tantos de nós falhamos em perder peso? Por que não conseguimos ir para a cama cedo e acordar cedo? Será por falta de determinação? De jeito nenhum. A questão é que estamos fazendo isso da maneira errada. Mais especificamente, é porque não
Ler
mais


10:33
1 de 5
A construção de hábitos requer quatro etapas: estímulo, desejo, resposta e recompensa são os pilares de todos os hábitos.
Falso
Verdadeiro

10:33
5 de 5
A Regra dos Dois Minutos é uma maneira rápida de acabar com a procrastinação, mas só funciona por dois minutos e faz pouco para construir hábitos de longo prazo.
Falso
Resposta correta
Uma vez que você aprendeu a cuidar da semente de cada hábito, os primeiros dois minutos são apenas o início das questões formais. Com o tempo, você esquecerá o limite de tempo de dois minutos e melhorará na construção do hábito.
Continuar

Capítulo 4 | Frases das páginas 253-326

1. A instrução SELECT cumpre essa função. Mas o SELECT pode fazer muito mais do que simplesmente recuperar e exibir dados.
2. Ter a habilidade de escolher colunas pode não parecer interessante no momento, mas nos permite focar no que realmente nos interessa.
3. Este é um recurso poderoso do SQL, que nos permite manter os dados armazenados simples e usar consultas para adicionar cálculos em cima disso.
4. Expressões não precisam trabalhar apenas com números. Você também pode usar expressões com texto e outros tipos de dados.
5. Ele recupera e transforma dados de uma tabela sem afetar a própria tabela.
6. Quando você começa a escrever condições WHERE complicadas, essa prática se torna ainda mais crítica.
7. Valores nulos não podem ser determinados com um =. Você precisa usar as instruções IS NULL ou IS NOT NULL para

Mais Livros Grátis
Escaneie para baixar
Ouça-o

identificar valores nulos.

8. Uma maneira mais elegante de lidar com valores nulos é usar a função coalesce(), que transformará um valor possivelmente nulo em um valor padrão especificado, caso seja nulo.
9. O SQL já está se mostrando útil. Você pode filtrar dados rápida e facilmente com condições muito explícitas, de uma forma que é difícil de conseguir em ferramentas do dia a dia como o Excel.
10. Começaremos a aprender mais funções e a usá-las na cláusula WHERE.

Capítulo 5 | Frases das páginas 327-464

1. Você pode filtrar dados de forma rápida e fácil com condições bem explícitas, de uma maneira que é difícil de alcançar em ferramentas do dia a dia como o Excel.
2. Agrupamentos com parênteses nas instruções WHERE não apenas tornam a semântica mais clara, mas também a execução mais segura.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

3. Funções agregadas como COUNT(), SUM(), AVG(), MIN() e MAX() nunca incluirão valores nulos em seus cálculos. Apenas valores não nulos serão considerados.
4. Enquanto seu primeiro instinto pode ser usar uma instrução WHERE, na verdade isso não funcionará porque o WHERE filtra registros e não filtra agregações.
5. Espero que agora você veja a flexibilidade que o SQL oferece para desenvolver rapidamente relatórios significativos com base em milhares ou milhões de registros.

Capítulo 6 | Frases das páginas 465-547

1. Agora vemos a contagem de avistamentos de tornados por ano.
2. Você não pode filtrar campos agregados usando WHERE. É necessário usar a palavra-chave HAVING para fazer isso.
3. O truque do CASE para zero/nulo é uma ótima utilização da instrução CASE. Ela oferece várias possibilidades para realizar diversas agregações com critérios diferentes e, portanto, vale a pena conhecer.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

4. Podemos fatiar esses dados em mais de um campo.
5. Espero que agora você veja a flexibilidade que o SQL oferece para desenvolver rapidamente relatórios significativos com base em milhares ou milhões de registros.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Escanear para baixar

Baixe o app Bookey para desfrutar

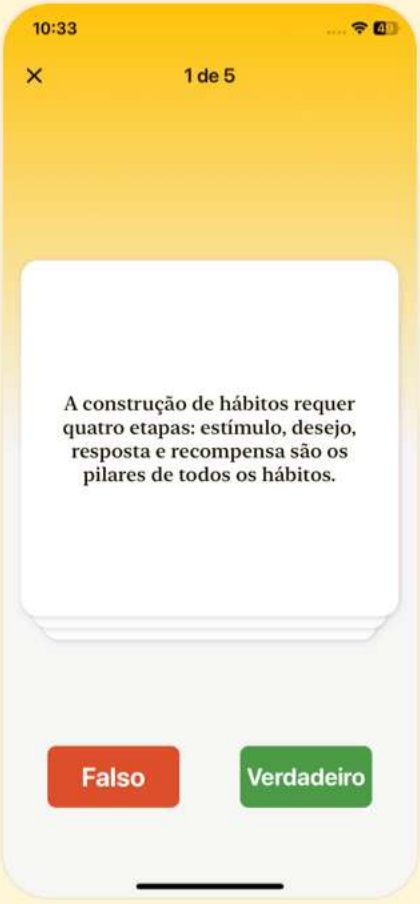
Mais de 1000 resumos de livros com quizzes

Teste grátis disponível!

Descarregar na App Store

OBTER NO Google Play

18:37
Hábitos Atômicos
Quatro passos para criar bons hábitos e quebrar maus costumes
James Clear
3 insights principais
Descrição
Por que tantos de nós falhamos em perder peso? Por que não conseguimos ir para a cama cedo e acordar cedo? Será por falta de determinação? De jeito nenhum. A questão é que estamos fazendo isso da maneira errada. Mais especificamente, é porque não
Ler
mais


10:33
1 de 5
A construção de hábitos requer quatro etapas: estímulo, desejo, resposta e recompensa são os pilares de todos os hábitos.
Falso
Verdadeiro

10:33
5 de 5
A Regra dos Dois Minutos é uma maneira rápida de acabar com a procrastinação, mas só funciona por dois minutos e faz pouco para construir hábitos de longo prazo.
Falso
Resposta correta
Uma vez que você aprendeu a cuidar da semente de cada hábito, os primeiros dois minutos são apenas o início das questões formais. Com o tempo, você esquecerá o limite de tempo de dois minutos e melhorará na construção do hábito.
Continuar

Capítulo 7 | Frases das páginas 548-615

1. Uma instrução CASE nos permite mapear uma ou mais condições a um valor correspondente para cada condição.
2. Saber disso nos permite criar uma consulta ligeiramente mais eficiente.
3. O truque do CASE zero/nulo é um ótimo uso da instrução CASE.
4. Joins nos permitem pegar dados espalhados por várias tabelas e costurá-los em algo mais significativo e descritivo.
5. Joins são o tópico mais desafiador em SQL, mas também são os mais recompensadores.

Capítulo 8 | Frases das páginas 616-758

1. Os joins nos permitem pegar dados espalhados por várias tabelas e juntá-los em algo mais significativo e descritivo.
2. Os joins realmente nos oferecem o melhor dos dois mundos. Armazenamos dados de forma eficiente através da

Mais Livros Grátis
Escaneie para baixar
Ouça-o

normalização, mas podemos usar joins para mesclar tabelas em campos comuns e criar visões mais descritivas dos dados.

3. Com um INNER JOIN, quaisquer registros que não tenham um valor comum conectado em ambas as tabelas serão excluídos.
4. Um LEFT JOIN incluirá todos os registros na tabela 'esquerda', mesmo que não tenham nada para se juntar na tabela 'direita' (que será nulo).
5. Os joins são o tópico mais desafiador em SQL, mas também são os mais gratificantes.

Capítulo 9 | Frases das páginas 759-1040

1. Você precisa planejar cuidadosamente, pois um mau design de banco de dados certamente causará arrependimentos no futuro.
2. Se os dados não forem mantidos e atualizados, então o design já falhou.
3. Segurança excessiva cria burocracia e obstrui a agilidade, mas segurança insuficiente convidará à calamidade.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

4. Uma chave primária é um campo especial (ou combinação de campos) que fornece uma identidade única a cada registro.
5. Um bom design com restrições bem definidas criará um banco de dados resiliente.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Escanear para baixar

Baixe o app Bookey para desfrutar

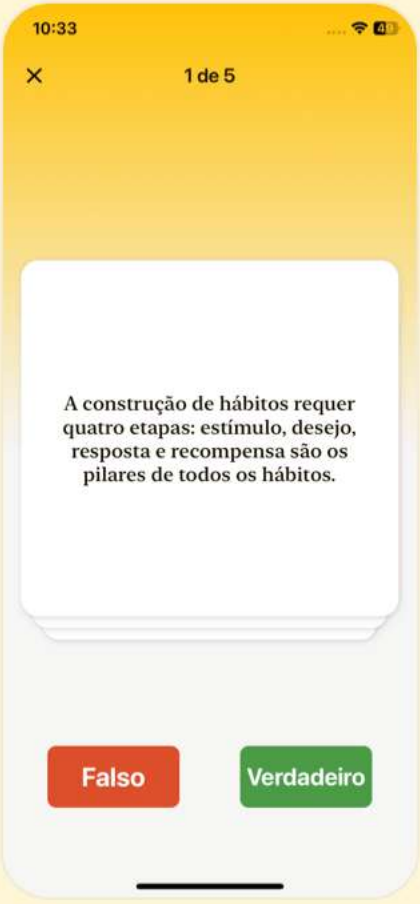
Mais de 1000 resumos de livros com quizzes

Teste grátis disponível!

Descarregar na App Store

OBTER NO Google Play

18:37
Hábitos Atômicos
Quatro passos para criar bons hábitos e quebrar maus costumes
James Clear
3 insights principais
Descrição
Por que tantos de nós falhamos em perder peso? Por que não conseguimos ir para a cama cedo e acordar cedo? Será por falta de determinação? De jeito nenhum. A questão é que estamos fazendo isso da maneira errada. Mais especificamente, é porque não
Ler
mais


10:33
1 de 5
A construção de hábitos requer quatro etapas: estímulo, desejo, resposta e recompensa são os pilares de todos os hábitos.
Falso
Verdadeiro

10:33
5 de 5
A Regra dos Dois Minutos é uma maneira rápida de acabar com a procrastinação, mas só funciona por dois minutos e faz pouco para construir hábitos de longo prazo.
Falso
Resposta correta
Uma vez que você aprendeu a cuidar da semente de cada hábito, os primeiros dois minutos são apenas o início das questões formais. Com o tempo, você esquecerá o limite de tempo de dois minutos e melhorará na construção do hábito.
Continuar

Capítulo 10 | Frases das páginas 1041-1117

1. Com um bom design de tabelas e relacionamentos bem pensados entre elas, seremos capazes de realizar junções de forma eficiente e fácil.
2. Quando um dado ruim entra, esperamos ter definido restrições sensatas o suficiente para impedir que ele entre no banco de dados.
3. Felizmente, operações como estas são muito mais simples do que escrever instruções SELECT.
4. Se um processo escrito em Java ou Python está populando uma tabela, ele deve usar esta sintaxe para inserir grandes quantidades de registros em vez de executar um INSERT por vez.
5. Isso é bom porque significa que nossa restrição de chave estrangeira funcionou: ela impediu que um registro órfão entrasse.
6. Porque é extremamente fácil cometer erros com uma instrução DELETE, é uma boa prática substituir o DELETE por um SELECT * primeiro.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

7. Neste ponto, você tem as ferramentas necessárias para criar seu próprio banco de dados e gerenciar seus dados.
8. Você pode ter perguntas sobre como fazer tudo isso de forma eficiente ou fornecer meios práticos para que os usuários insiram e atualizem dados.
9. Se você quiser aprender mais funcionalidade detalhadas em SQL, há muitos livros e recursos disponíveis.
10. Integrar algumas habilidades tecnológicas pode torná-lo ainda mais competitivo do que apenas se especializar em uma única habilidade.

Capítulo 11 | Frases das páginas 1118-1147

1. Parabéns! Você aprendeu um conjunto de habilidades adaptável e valioso que pode ser aplicado nas áreas de negócios, tecnologia da informação e engenharia.
2. Você só pode definir seus objetivos de carreira e priorizar o que maximiza seu valor!
3. Se você quiser aprender mais funcionalidades detalhadas em SQL, há muitos livros e recursos disponíveis.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

4. Integrar algumas habilidades tecnológicas pode tornar você ainda mais valorizado do que apenas se especializar em uma única habilidade.
5. As chances são de que se você tem uma dúvida ou um problema, outra pessoa provavelmente também já teve e a resposta foi posta online.
6. Essas pessoas oferecem expertise simplesmente porque são apaixonadas pelo que fazem.

Capítulo 12 | Frases das páginas 1148-1279

1. Você pode testar operadores e funções facilmente, sem consultar nenhuma tabela. Você simplesmente SELECIONA uma expressão de literais, como na seguinte consulta, que calculará um único valor de 12: SELECIONE 5 + 7
2. O SQLite possui um pequeno conjunto de operadores matemáticos básicos. Tarefas mais avançadas geralmente são realizadas com funções...
3. Os operadores de comparação retornam um valor verdadeiro (1) ou falso (0) com base em uma avaliação

Mais Livros Grátis
Escaneie para baixar
Ouça-o

comparativa.

4. Os operadores lógicos permitem combinar expressões booleanas, bem como realizar operações condicionais mais complexas.
5. O SQLite possui muitas funções principais integradas. Embora esta não seja uma lista abrangente, estas são as mais comumente usadas.
6. As transações executam múltiplas instruções UPDATE/DELETE/INSERT como um único lote de segurança.
7. Você deve evitar usar índices em tabelas pequenas, pois a sobrecarga realmente diminuirá o desempenho.
8. Você vai querer usar [transações] especialmente com processos automatizados ou grandes trabalhos que exigem que você INSIRA, ATUALIZE e DELETE um alto volume de registros frágeis.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Escanear para baixar

Baixe o app Bookey para desfrutar

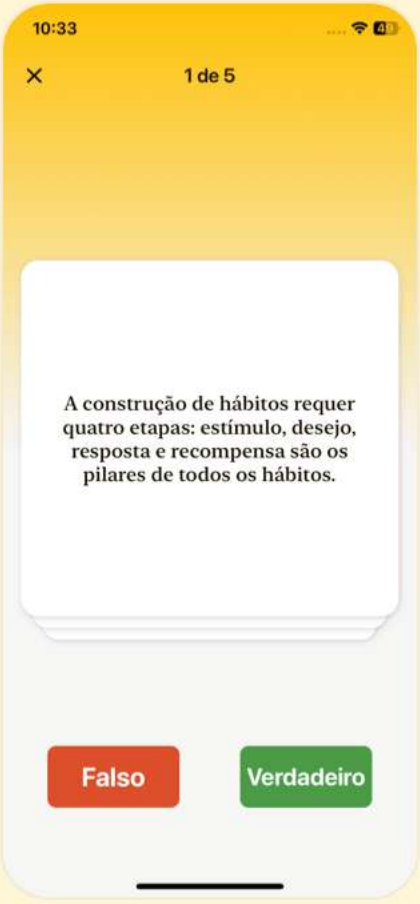
Mais de 1000 resumos de livros com quizzes

Teste grátis disponível!

Descarregar na App Store

OBTER NO Google Play

18:37
Hábitos Atômicos
Quatro passos para criar bons hábitos e quebrar maus costumes
James Clear
3 insights principais
Descrição
Por que tantos de nós falhamos em perder peso? Por que não conseguimos ir para a cama cedo e acordar cedo? Será por falta de determinação? De jeito nenhum. A questão é que estamos fazendo isso da maneira errada. Mais especificamente, é porque não
Ler
mais


10:33
1 de 5
A construção de hábitos requer quatro etapas: estímulo, desejo, resposta e recompensa são os pilares de todos os hábitos.
Falso
Verdadeiro

10:33
5 de 5
A Regra dos Dois Minutos é uma maneira rápida de acabar com a procrastinação, mas só funciona por dois minutos e faz pouco para construir hábitos de longo prazo.
Falso
Resposta correta
Uma vez que você aprendeu a cuidar da semente de cada hábito, os primeiros dois minutos são apenas o início das questões formais. Com o tempo, você esquecerá o limite de tempo de dois minutos e melhorará na construção do hábito.
Continuar

Capítulo 13 | Frases das páginas 1280-1343

1. Provavelmente você encontrará dezenas de outros tópicos, especialmente à medida que explora as nuances de diferentes plataformas de banco de dados. Mas estes devem fornecer muitas pistas para expandir seu conhecimento sobre bancos de dados além deste livro.
2. Você não deve pensar em criar índices quando projetar um banco de dados pela primeira vez. Faça isso mais tarde, quando perceber que está enfrentando problemas de desempenho.
3. Com uma transação, você pode tornar essa transferência atômica e fazer um ROLLBACK se algo falhar e um COMMIT se tudo foi bem-sucedido.
4. Na eventualidade de uma falha na conexão com o banco de dados, se uma instrução SQL ruim for composta ou se uma regra de validação causar um desastre em uma série de atualizações, as transações são uma maneira de garantir que os dados não sejam corrompidos em seu banco de dados.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Escanear para baixar

Baixe o app Bookey para desfrutar

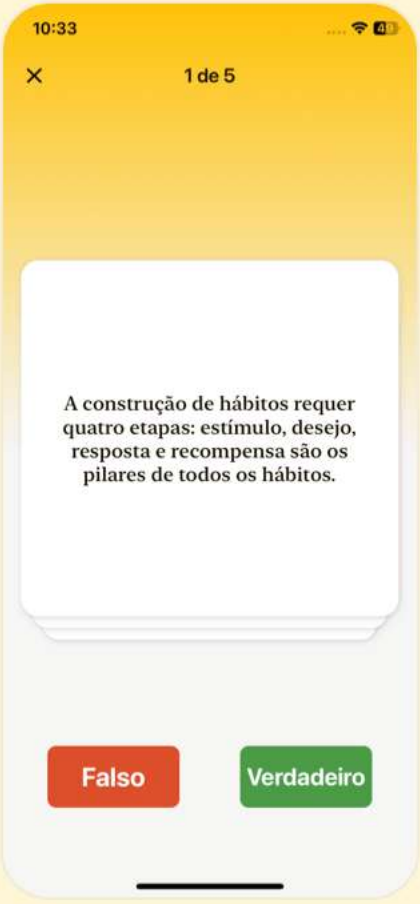
Mais de 1000 resumos de livros com quizzes

Teste grátis disponível!

Descarregar na App Store

OBTER NO Google Play

18:37
Hábitos Atômicos
Quatro passos para criar bons hábitos e quebrar maus costumes
James Clear
3 insights principais
Descrição
Por que tantos de nós falhamos em perder peso? Por que não conseguimos ir para a cama cedo e acordar cedo? Será por falta de determinação? De jeito nenhum. A questão é que estamos fazendo isso da maneira errada. Mais especificamente, é porque não
Ler
mais


10:33
1 de 5
A construção de hábitos requer quatro etapas: estímulo, desejo, resposta e recompensa são os pilares de todos os hábitos.
Falso
Verdadeiro

10:33
5 de 5
A Regra dos Dois Minutos é uma maneira rápida de acabar com a procrastinação, mas só funciona por dois minutos e faz pouco para construir hábitos de longo prazo.
Falso
Resposta correta
Uma vez que você aprendeu a cuidar da semente de cada hábito, os primeiros dois minutos são apenas o início das questões formais. Com o tempo, você esquecerá o limite de tempo de dois minutos e melhorará na construção do hábito.
Continuar

Introduçao a Linguagem SQL Perguntas

Ver no site do Bookey

Capítulo 1 | 1. Por Que Aprender SQL?| Perguntas e respostas

1.Pergunta

Qual é a principal importância do SQL no cenário de negócios atual?

Resposta:SQL, ou Linguagem de Consulta Estrutural, serve como uma ferramenta crucial para acessar e manipular grandes volumes de dados gerados pelas empresas. À medida que as organizações dependem cada vez mais de dados para impulsionar a tomada de decisões e a estratégia, o SQL permite que os profissionais derive insights significativos desses dados e ajuda a conectar os dados brutos à inteligência de negócios acionável.

2.Pergunta

Quem se beneficia ao aprender SQL e por que é uma habilidade valiosa em diferentes profissões?

Mais Livros Grátis

Escaneie para baixar

Ouça-o

Resposta:O SQL não se limita apenas a profissionais de TI; é valioso para empresários, gerentes, engenheiros e analistas. Aprender SQL capacita os indivíduos a obter insights mais profundos sobre os dados de sua organização, abrindo diversas oportunidades de carreira em áreas como análise, gerenciamento de projetos e sistemas de TI. Essa versatilidade torna-o uma habilidade essencial em muitos campos.

3.Pergunta

Como a demanda por habilidades em SQL reflete o atual mercado de trabalho?

Resposta:À medida que os dados se tornaram ubíquos, existe uma crescente lacuna entre a quantidade de dados coletados e o número de profissionais que podem analisá-los e gerenciá-los efetivamente. Essa demanda posiciona o SQL como uma habilidade altamente comercializável, com salários competitivos refletindo sua importância em vários papéis em diversas indústrias.

4.Pergunta

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Qual é a importância da normalização de banco de dados e como ela se relaciona com o SQL?

Resposta:A normalização de banco de dados é vital, pois minimiza a redundância de dados e melhora a integridade dos dados. No SQL, esse conceito fundamenta como as tabelas são estruturadas e relacionadas—assegurando que cada tabela contenha dados únicos sem duplicação desnecessária. Por exemplo, separar as informações dos clientes em sua própria tabela garante que mudanças, como atualizações de endereço, precisem ser feitas apenas em um local, promovendo a eficiência.

5.Pergunta

O que são bancos de dados leves e centralizados, e como eles diferem em uso?

Resposta:Bancos de dados leves, como o SQLite, são projetados para conjuntos de dados menores ou usuários únicos, exigindo configuração e sobrecarga mínimas. Em contraste, bancos de dados centralizados como MySQL ou Microsoft SQL Server suportam múltiplos usuários e

Mais Livros Grátis
Escaneie para baixar
Ouça-o

aplicações de alto tráfego, operando em servidores dedicados para gerenciar grandes volumes de dados de forma eficiente. Compreender quando usar cada tipo é fundamental, dependendo da escala e das exigências da aplicação do banco de dados.

6.Pergunta

Por que o SQL é considerado uma linguagem universal para interações com bancos de dados?

Resposta:O SQL é considerado uma linguagem universal porque oferece uma maneira padronizada de interagir com diferentes sistemas de banco de dados, permitindo que os usuários realizem consultas, insiram dados e gerenciem bancos de dados de forma consistente em várias plataformas. Embora diferentes bancos de dados possam implementar o SQL com algumas variações, as funcionalidades principais permanecem familiares, tornando o SQL uma habilidade essencial para qualquer pessoa que trabalhe com dados.

7.Pergunta

Quais são as implicações de carreira de aprender SQL no ambiente orientado a dados de hoje?

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Resposta:Adquirir habilidades em SQL pode aumentar significativamente a empregabilidade de uma pessoa, uma vez que as empresas estão buscando cada vez mais profissionais que compreendam dados e possam utilizá-los para estratégia e insights. Essa demanda se traduz em várias potenciais oportunidades de carreira, desde analistas de dados e administradores de banco de dados até funções que exigem gestão e interpretação de dados, expandindo as oportunidades de carreira em setores de TI e negócios.

Capítulo 2 | 2. Bancos de Dados| Perguntas e respostas

1.Pergunta

O que é um banco de dados em um sentido amplo?

Resposta:Um banco de dados é qualquer coisa que coleta e organiza dados, como planilhas e arquivos de texto simples.

2.Pergunta

O que é um sistema de gerenciamento de banco de dados relacional (RDBMS)?

Resposta:Um RDBMS é um tipo de banco de dados que

Mais Livros Grátis
Escaneie para baixar
Ouça-o

contém uma ou mais tabelas que podem ter relacionamentos entre si.

3.Pergunta

Por que as tabelas em um banco de dados relacional são separadas?

Resposta:A motivação para separar tabelas é a normalização. Isso evita redundância, mantém os dados gerenciáveis e simplifica a manutenção, como a atualização das informações de clientes.

4.Pergunta

Como a normalização impacta a gestão de dados?

Resposta:A normalização reduz redundâncias; por exemplo, se o endereço de um cliente mudar, ele precisa ser atualizado em um único lugar em vez de várias entradas em uma única tabela.

5.Pergunta

O que são bancos de dados leves e quando são apropriados?

Resposta:Bancos de dados leves, como SQLite e Microsoft Access, são adequados para casos de uso simples com

Mais Livros Grátis
Escaneie para baixar
Ouça-o

poucos usuários. Eles são fáceis de configurar, tornando-os ideais para projetos pessoais ou pequenas equipes.

6.Pergunta

O que é SQLite e por que é usado?

Resposta:SQLite é um mecanismo de banco de dados leve amplamente utilizado em aplicativos móveis e embarcados devido à sua simplicidade, baixo consumo de recursos e facilidade de uso.

7.Pergunta

Quais são os benefícios de usar SQLite para aprender SQL?

Resposta:SQLite é ideal para aprender SQL devido à sua facilidade de instalação, simplicidade na configuração e sua presença em vários dispositivos, tornando-o familiar aos alunos.

8.Pergunta

Qual é a diferença entre bancos de dados leves e bancos de dados centralizados?

Resposta:Bancos de dados leves são adequados para poucos usuários e têm um consumo mínimo de recursos, enquanto

Mais Livros Grátis
Escaneie para baixar
Ouça-o

bancos de dados centralizados são projetados para ambientes de alto tráfego que suportam muitos usuários simultâneos.

9.Pergunta

Qual é a importância do SQL em diferentes soluções de banco de dados?

Resposta:O SQL é a linguagem universal para interagir com bancos de dados, permitindo que os usuários consultem e manipulem dados de maneira consistente em diferentes plataformas.

10.Pergunta

Por que as organizações podem precisar migrar de bancos de dados leves para bancos de dados centralizados?

Resposta:À medida que o número de usuários ou o volume de dados aumenta, os bancos de dados leves podem ter dificuldades, necessitando de uma solução centralizada mais robusta para gerenciar os dados de forma eficiente.

Capítulo 3 | 3. SQLite| Perguntas e respostas

1.Pergunta

O que torna o SQLite uma escolha adequada para

Mais Livros Grátis
Escaneie para baixar
Ouça-o

armazenamento de dados em aplicações modernas?

Resposta:A simplicidade e o baixo overhead do SQLite o tornam ideal para aplicações que requerem gerenciamento leve de dados. Ele é embutido em muitos dispositivos (como smartphones, tablets e eletrodomésticos) onde um sistema de arquivos simples é necessário em vez de uma configuração cliente-servidor. Seu uso generalizado em sistemas como Windows 10 e Airbus A350 destaca sua eficiência.

2.Pergunta

Como você pode importar bancos de dados no SQLiteStudio?

Resposta:Para importar um banco de dados no SQLiteStudio, primeiro baixe os arquivos do banco de dados que deseja usar. Em seguida, no SQLiteStudio, navegue até Banco de Dados !’ Adicionar um Banco de Dados. Se do banco de dados que você baixou para carregá-lo no aplicativo. Agora você pode explorar seu conteúdo.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

3.Pergunta

Por que entender os tipos de dados é importante ao trabalhar com bancos de dados?

Resposta:Entender os tipos de dados é crucial porque isso dita como os dados são armazenados e manipulados. Por exemplo, campos definidos como INTEGER devem conter números inteiros, enquanto campos TEXT armazenam valores de string. Saber disso ajuda a evitar erros e permite consultas de dados eficazes.

4.Pergunta

Qual é a importância da instrução SELECT em SQL?

Resposta:A instrução SELECT é o comando fundamental em SQL usado para solicitar dados de bancos de dados. Ele não apenas recupera dados, mas também pode transformá-los por meio de cálculos e formatações, permitindo que os usuários analisem e resumam grandes conjuntos de dados de forma eficiente.

5.Pergunta

Você pode explicar o uso de expressões em instruções SELECT e fornecer um exemplo?

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Resposta:Expressões em instruções SELECT permitem cálculos sobre dados retornados por consultas. Por exemplo, se você tiver uma tabela de PRODUTO e quiser calcular um PREÇO_COM_IMPOSTO que seja 7% maior que o preço original, você pode escrever: SELECT PRODUCT_ID, DESCRIPTION, PRICE, PRICE * 1.07 AS PREÇO_COM_IMPOSTO FROM PRODUCT; Isso calcula dinamicamente o PREÇO_COM_IMPOSTO para cada produto sempre que a consulta é executada.

6.Pergunta

Qual é o propósito dos aliases em SQL e como eles são usados?

Resposta:Aliases são usados em SQL para dar nomes temporários a colunas ou tabelas, tornando a saída mais legível. Por exemplo, na consulta: SELECT PRICE AS PREÇO_SEM_IMPOSTO, PRICE * 1.07 AS PREÇO_COM_IMPOSTO FROM PRODUCT; as colunas mostrarão os rótulos 'PREÇO_SEM_IMPOSTO' e 'PREÇO_COM_IMPOSTO' em vez de apenas os nomes de

Mais Livros Grátis
Escaneie para baixar
Ouça-o

colunas padrão.

7.Pergunta

Como o SQLite trata a concatenação e por que isso é útil?

Resposta:O SQLite usa o operador pipe duplo (||) para concatenar strings. Isso é útil para combinar vários campos em uma única saída, como unir a cidade e o estado de um cliente em um valor de LOCALIZAÇÃO. Por exemplo, concatenando CIDADE e ESTADO: SELECT NAME, CITY || ', ' || STATE AS LOCALIZAÇÃO FROM CUSTOMER; permite exibir um endereço consolidado.

8.Pergunta

Quais são alguns operadores matemáticos suportados em SQL e quando você os usaria?

Resposta:O SQL suporta vários operadores matemáticos, incluindo + para adição, - para subtração, * para multiplicação e / para divisão. Estes são tipicamente usados em cálculos dentro de instruções SELECT para processar dados dinamicamente, como calcular totais ou médias.

9.Pergunta

O que você pode fazer para melhorar a apresentação de

Mais Livros Grátis
Escaneie para baixar
Ouça-o

saídas numéricas em consultas SQL?

Resposta:Para melhorar saídas numéricas, você pode usar a função round() para limitar o número de casas decimais exibidas. Por exemplo: SELECT round(PRICE * 1.07, 2) AS PREÇO_COM_IMPOSTO FROM PRODUCT; Isso garante que o preço seja exibido em um formato mais amigável ao usuário.

10.Pergunta

O que será discutido no próximo capítulo após a instrução SELECT?

Resposta:O próximo capítulo apresentará a instrução WHERE, que permite aos usuários filtrar registros com base em critérios especificados, aprimorando ainda mais as capacidades de consulta de dados.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Ad

Escanear para baixar

11:12
Capítulo 1 de 5
Visão Geral
Olá, bem-vindo ao Bookey. Hoje vamos explorar o livro Comunicação Não Violenta: Uma Linguagem de Vida.
Nós nos comunicamos com os outros todos os dias. Se prestarmos mais atenção às maneiras como falamos e aos nossos sentimentos em relação a várias formas de conversa, perceberemos que algumas palavras podem ser extremamente prejudiciais. Por exemplo, há negações verbais, interrupções rudes e julgamentos casuais. Agora, o livro Comunicação Não Violenta: Uma Linguagem de Vida nos ensinará uma nova maneira de comunicação, permitindo-nos evitar tais danos e obter mais amor e felicidade
11:12
Comunicação Não Violenta
Uma linguagem da vida
Marshall Rosenberg
Descrição
Ler

Por que o Bookey é um aplicativo indispensável para amantes de livros

Conteúdo de 30min
Quanto mais profunda e clara for a interpretação que fornecemos, melhor será sua compreensão de cada título.

Clipes de Ideias de 3min
Impulsione seu progresso.

Questionário
Verifique se você dominou o que acabou de aprender.

E mais
Várias fontes, Caminhos em andamento, Coleções...

Teste gratuito com Bookey

Capítulo 4 | 4. SELECT| Perguntas e respostas

1.Pergunta

Qual é o propósito da instrução SELECT em SQL?

Resposta:A instrução SELECT é usada para solicitar dados de uma ou mais tabelas em um banco de dados. Ela não apenas recupera e exibe dados, mas também permite transformações, cálculos e a criação de resumos com base em milhões de registros.

2.Pergunta

Como você pode escolher colunas específicas para recuperar de uma tabela usando SELECT?

Resposta:Em uma instrução SELECT, você especifica as colunas desejadas nomeando-as após a palavra-chave SELECT. Por exemplo, para recuperar apenas as colunas CUSTOMER_ID e NAME da tabela CUSTOMER, você usaria:

SELECT CUSTOMER_ID, NAME FROM CUSTOMER;.

3.Pergunta

O que são expressões em instruções SELECT e por que

Mais Livros Grátis
Escaneie para baixar
Ouça-o

são úteis?

Resposta:Expressões em instruções SELECT permitem que você realize cálculos sobre os dados. Por exemplo, calcular um PREÇO_COM_IMPOSTO como 7% maior que o PREÇO permite manipulação dinâmica de dados diretamente na consulta, sem alterar os dados originais, aumentando a capacidade de criar cálculos em tempo real.

4.Pergunta

Explique como usar alias em SQL e por que eles são úteis.

Resposta:Aliases em SQL são criados usando a palavra-chave AS e são usados para dar nomes temporários a colunas ou expressões para clareza ou conveniência. Por exemplo, usando:

SELECT PRICE AS PREÇO_SEM_IMPOSTO, PRICE * 1.07 AS PREÇO_COM_IMPOSTO FROM PRODUCT;

eso torna os resultados mais compreensíveis sem modificar a estrutura subjacente da tabela.

5.Pergunta

Qual função você pode usar para limitar casas decimais

Mais Livros Grátis
Escaneie para baixar
Ouça-o

em cálculos SQL?

Resposta:A função round() é usada para limitar o número de casas decimais em cálculos SQL. Você pode especificar o número a ser arredondado e a quantidade de casas decimais, o que ajuda a apresentar resultados mais limpos nas saídas.

6.Pergunta

Como funciona a concatenação de texto em SQL?

Resposta:A concatenação de texto em SQL une várias strings usando o operador de dupla barra (||). Por exemplo, combinando CIDADE e ESTADO para criar uma LOCALIZAÇÃO completa:

SELECT CITY || ', ' || STATE AS LOCATION;

Esse recurso permite apresentações de dados mais informativas e compreensíveis.

7.Pergunta

Qual é a importância de filtrar registros em SQL usando a cláusula WHERE?

Resposta:Filtrar registros usando a cláusula WHERE permite uma recuperação precisa de dados com base em critérios

Mais Livros Grátis
Escaneie para baixar
Ouça-o

específicos, habilitando os usuários a extrair apenas informações relevantes de grandes conjuntos de dados.

8.Pergunta

Como você pode filtrar faixas específicas de números em SQL?

Resposta:Você pode usar o operador BETWEEN ou declarações AND para filtrar faixas específicas de números. Por exemplo, para recuperar registros de anos entre 2005 e 2010 (inclusivo), você escreveria:

SELECT * FROM station_data WHERE year BETWEEN 2005 AND 2010;.

9.Pergunta

Qual é a diferença entre valores NULL e strings vazias em SQL?

Resposta:Valores NULL representam a ausência de qualquer valor ou conteúdo, enquanto strings vazias (" ou ' ') são valores reais que indicam a falta de conteúdo. NULL é tratado de maneira diferente e muitas vezes precisa de um tratamento explícito em consultas SQL.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

10.Pergunta

Por que é importante agrupar condições em consultas SQL?

Resposta:Agrupar condições usando parênteses esclarece a estrutura lógica da consulta e garante a aplicação correta dos operadores AND e OR. Isso previne ambiguidade em declarações complexas, assim como seguir a ordem de operações na matemática.

Capítulo 5 | 5. WHERE| Perguntas e respostas

1.Pergunta

Como a cláusula WHERE melhora a filtragem de dados ao trabalhar com SQL?

Resposta:A cláusula WHERE permite que os usuários filtrem registros com base em critérios específicos, garantindo que apenas dados relevantes sejam retornados. Usando expressões como '=', '!=', 'BETWEEN' e operadores lógicos como 'AND', 'OR' e 'IN', os usuários podem direcionar suas consultas de dados de forma eficaz, tornando a

Mais Livros Grátis
Escaneie para baixar
Ouça-o

análise de dados mais eficiente.

2.Pergunta

O que o comando BETWEEN faz em instruções SQL?

Resposta:O comando BETWEEN é utilizado para filtrar registros por uma faixa de valores, onde ambos os limites estão inclusos nos resultados. Por exemplo, usando 'WHERE year BETWEEN 2005 AND 2010' seleciona registros para os anos de 2005 a 2010.

3.Pergunta

Como você pode evitar ambiguidades em consultas SQL complexas com múltiplas condições?

Resposta:Para evitar ambiguidades em consultas SQL complexas, é importante usar parênteses para agrupar condições logicamente. Por exemplo, em uma consulta com operadores AND e OR, o uso de parênteses clareia quais condições estão relacionadas e garante que o SQL as processe corretamente.

4.Pergunta

Qual é a importância de usar funções agregadas como COUNT e SUM em SQL?

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Resposta:As funções agregadas, como COUNT e SUM, permitem que os usuários realizem cálculos em grupos de dados, resumindo ou acumulando valores com base em critérios especificados. Por exemplo, COUNT pode determinar o número de ocorrências de uma condição, enquanto SUM pode calcular o total de uma coluna numérica.

5.Pergunta

Como você lida com valores NULL em SQL ao executar consultas?

Resposta:Os valores NULL podem ser tratados usando as condições IS NULL ou IS NOT NULL em SQL, permitindo que os usuários filtrem ou incluam registros com dados ausentes. Além disso, funções como coalesce() podem fornecer valores padrão no lugar de NULLs, garantindo que as consultas funcionem corretamente mesmo com dados incompletos.

6.Pergunta

Qual é o propósito da cláusula HAVING em SQL?

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Resposta:A cláusula HAVING é usada para filtrar os resultados de operações de agregação realizadas em uma consulta SQL. Diferentemente do WHERE, que filtra registros antes da agregação, o HAVING filtra os dados após a agregação, tornando possível incluir apenas grupos que atendem a certas condições.

7.Pergunta

Por que é importante distinguir entre valores NULL e strings vazias em SQL?

Resposta:É crucial distinguir entre valores NULL, que representam a ausência de qualquer dado, e strings vazias, que são consideradas valores válidos. Confundir os dois pode levar a resultados de consulta incorretos, uma vez que os NULLs são automaticamente filtrados de operações, enquanto strings vazias são tratadas como dados reais.

8.Pergunta

Como você pode recuperar valores distintos de um conjunto de dados usando SQL?

Resposta:Para recuperar valores distintos de um conjunto de

Mais Livros Grátis
Escaneie para baixar
Ouça-o

dados, você pode usar a palavra-chave DISTINCT em sua instrução SELECT. Por exemplo, 'SELECT DISTINCT station_number FROM station_data' retornará uma lista de números de estações únicos, sem duplicatas.

9.Pergunta

O que você deve fazer se uma função agregada não estiver incluindo valores NULL em seus cálculos?

Resposta:Funções agregadas como SUM, COUNT e AVG automaticamente excluem valores NULL de seus cálculos. Para analisar contagens ou médias onde os NULLs são importantes, você pode contar especificamente os valores não nulos usando 'COUNT(column_name)' ou lidar com NULLs convertendo-os em um valor padrão usando métodos como a função coalesce().'

10.Pergunta

Que tipo de perguntas você pode explorar usando consultas SQL com base no conteúdo fornecido?

Resposta:Você pode explorar várias questões analíticas com SQL, como tendências ao longo do tempo (por exemplo, 'A

Mais Livros Grátis
Escaneie para baixar
Ouça-o

temperatura tem ficado mais quente a cada janeiro nos últimos 20 anos?'), comparações (por exemplo, 'Quantas vezes o granizo esteve presente em comparação a quando não esteve durante um tornado?') ou calcular estatísticas agregadas (por exemplo, 'Qual é a precipitação média em cada mês desde 2000?')

Capítulo 6 | 6. GROUP BY e ORDER BY| Perguntas e respostas

1.Pergunta

Qual é a importância do uso da cláusula GROUP BY no SQL?

Resposta:A cláusula GROUP BY é essencial para a agregação de dados no SQL. Ela permite agrupar linhas que têm os mesmos valores em colunas especificadas em linhas resumo, como contar o número de ocorrências ou agregar valores (por exemplo, soma, média) entre esses grupos. Isso é particularmente útil para produzir análises significativas a partir de grandes conjuntos de dados, como calcular totais por ano ou mês.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

2.Pergunta

Como você pode fazer um relatório significativo usando as funções de agregação do SQL?

Resposta:Usando funções de agregação como COUNT(), SUM(), AVG(), MIN() e MAX() em combinação com o GROUP BY, você pode resumir conjuntos de dados em relatórios significativos. Por exemplo, você poderia contar o número de avistamentos de tornados por ano para identificar tendências ao longo do tempo, ou calcular as temperaturas médias por mês para avaliar mudanças climáticas.

3.Pergunta

Qual é o propósito da cláusula HAVING no SQL?

Resposta:A cláusula HAVING é usada para filtrar registros após as agregações terem ocorrido, ao contrário da cláusula WHERE, que filtra antes da agregação. Isso significa que você pode usar HAVING quando deseja aplicar condições nos resultados das funções de agregação, como filtrar grupos onde a precipitação total excede um certo valor.

4.Pergunta

Como o uso da função COUNT() em uma coluna difere

Mais Livros Grátis
Escaneie para baixar
Ouça-o

do uso com um asterisco?

Resposta:Quando COUNT() é usado com um asterisco (*), ele conta todos os registros em uma tabela, incluindo aqueles com valores NULL. No entanto, quando é utilizado com um nome de coluna específico, ele conta apenas os valores não nulos naquela coluna, o que pode ser útil para determinar quantos registros têm dados válidos.

5.Pergunta

Você pode explicar o truque 'zero/null' em SQL?

Resposta:O truque 'zero/null' permite criar lógica condicional que categoriza e agrega dados de forma eficaz em uma única consulta. Em vez de escrever várias consultas ou usar WHERE para filtrar, você pode usar uma declaração CASE dentro de uma função de agregação, atribuindo 0 ou NULL com base em condições. Isso ajuda a somar ou calcular médias de valores seletivamente com base em critérios, auxiliando em análises de dados mais complexas.

6.Pergunta

Como a ordem das condições em uma declaração CASE

Mais Livros Grátis
Escaneie para baixar
Ouça-o

afeta sua avaliação?

Resposta:Em uma declaração CASE, o SQL avalia as condições na ordem em que são escritas. A primeira condição verdadeira que encontra determina o valor de saída. Portanto, a ordem é muito importante; se uma condição mais alta for colocada antes de uma mais baixa, ela será avaliada primeiro, e os intervalos mais baixos subsequentes não serão verificados, mesmo que sejam verdadeiros para aquele registro.

7.Pergunta

Que insights você pode derivar da análise da temperatura média usando declarações CASE?

Resposta:Ao segmentar as temperaturas médias com base na presença de tipos de precipitação como chuva ou granizo através de declarações CASE, você pode entender como as condições climáticas afetam as temperaturas. Isso pode informar estudos sobre padrões climáticos ou auxiliar no planejamento agrícola, mostrando como a temperatura se comporta de maneira diferente sob vários cenários de

Mais Livros Grátis
Escaneie para baixar
Ouça-o

precipitação.

8.Pergunta

Quais podem ser as implicações de usar funções de agregação sem agrupamento?

Resposta:Usar funções de agregação sem GROUP BY pode levar a totais gerais para todo o conjunto de dados, em vez de insights sobre dados segmentados. Isso pode ocultar tendências e padrões que afetam diferentes categorias, o que pode levar a conclusões enganosas. O uso adequado do GROUP BY permite uma análise direcionada onde variações significativas podem ser destacadas.

9.Pergunta

Em quais situações seria essencial usar a palavra-chave DISTINCT no SQL?

Resposta:A palavra-chave DISTINCT é essencial quando você precisa recuperar valores únicos de uma coluna ou combinação de colunas, removendo duplicatas do conjunto de resultados. Isso é particularmente útil ao analisar conjuntos de dados onde a redundância pode obscurecer

Mais Livros Grátis
Escaneie para baixar
Ouça-o

contagens ou características reais, como identificar números de estações distintos ou eventos climáticos únicos.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Ad

Escanear para baixar

App Store
Escolha dos Editores

22k avaliações de 5 estrelas

Feedback Positivo

Afonso Silva

s cada resumo de livro não só
o, mas também tornam o
m divertido e envolvente. O
tou a leitura para mim.

Fantástico! Brígida Santos

Estou maravilhado com a variedade de livros e idiomas que o Bookey suporta. Não é apenas um aplicativo, é um portal para o conhecimento global. Além disso, ganhar pontos para caridade é um grande bônus!

Fi
O
só
o
O

na Oliveira

correr as
ém me dá
omprar a
ar!

Adoro! Duarte Costa

Usar o Bookey ajudou-me a cultivar um hábito de leitura sem sobrecarregar minha agenda. O design do aplicativo e suas funcionalidades são amigáveis, tornando o crescimento intelectual acessível a todos.

Economiza tempo!

O Bookey é o meu apli
crescimento intelectua
perspicazes e lindame
um mundo de conheci

Aplicativo incrível! Estevão Pereira

Eu amo audiolivros, mas nem sempre tenho tempo para ouvir o livro inteiro! O Bookey permite-me obter um resumo dos destaques do livro que me interessa!!! Que ótimo conceito!!! Altamente recomendado!

Aplicativo lindo

Este aplicativo é um salva-vidas para
de livros com agendas lotadas. Os re
precisos, e os mapas mentais ajudar
o que aprendi. Altamente recomend

Teste gratuito com Bookey

Capítulo 7 | 7. Instruções CASE| Perguntas e respostas

1.Pergunta

Qual é o propósito fundamental da instrução CASE em SQL?

Resposta:A instrução CASE nos permite trocar o valor de uma coluna por outro valor com base em condições especificadas. Ela proporciona flexibilidade para categorizar ou transformar dados dinamicamente dentro de uma consulta.

2.Pergunta

Como a ordem das condições dentro de uma instrução CASE afeta sua avaliação?

Resposta:A instrução CASE avalia as condições de cima para baixo, parando na primeira condição verdadeira. Isso significa que condições mais específicas devem ser colocadas antes das mais amplas para garantir uma categorização correta.

3.Pergunta

Explique a importância de agrupar instruções CASE em

Mais Livros Grátis
Escaneie para baixar
Ouça-o

consultas SQL?

Resposta:Agrupar instruções CASE permite poderosas transformações de dados antes da agregação. Isso nos permite analisar e categorizar dados dinamicamente, proporcionando insights mais profundos e tornando os dados mais gerenciáveis.

4.Pergunta

Qual é o truque 'zero/nulo' na instrução CASE e como ele é aplicado?

Resposta:O truque 'zero/nulo' na instrução CASE permite diferentes filtros para valores agregados dentro de uma única consulta usando lógica condicional. Em vez de filtrar com WHERE, você usa CASE para retornar um zero (ou NULL) se uma condição não for atendida, facilitando a análise simultânea de múltiplas condições.

5.Pergunta

Quais são os dois tipos básicos de operações JOIN e como eles diferem?

Resposta:INNER JOIN inclui apenas registros onde há

Mais Livros Grátis
Escaneie para baixar
Ouça-o

correspondência em ambas as tabelas unidas, enquanto LEFT JOIN inclui todos os registros da tabela da esquerda e correspondências da tabela da direita, com NULL para registros não correspondentes.

6.Pergunta

Por que é essencial aplicar JOINs corretamente em consultas SQL?

Resposta:JOINs aplicados corretamente nos permitem unir dados de diferentes tabelas, criando conjuntos de dados mais abrangentes para análise, orientando melhores decisões de negócios e descobrindo relações entre pontos de dados.

7.Pergunta

Descreva uma aplicação prática de juntar múltiplas tabelas em uma consulta.

Resposta:Uma aplicação prática seria recuperar pedidos de clientes juntamente com detalhes dos clientes e informações de produtos, unindo as tabelas CUSTOMER_ORDER, CUSTOMER e PRODUCT. Isso permite relatórios detalhados sobre vendas, demografia dos clientes e

Mais Livros Grátis
Escaneie para baixar
Ouça-o

especificações de produtos.

8.Pergunta

Como agregações condicionais podem ser realizadas usando instruções CASE?

Resposta:Agregações condicionais podem ser feitas usando CASE dentro de funções agregadas como SUM ou AVG. Você cria uma instrução CASE que define as condições e a usa para somar ou calcular a média de valores condicionalmente com base nesses critérios.

9.Pergunta

O que acontece com os registros ao usar INNER JOIN em comparação com LEFT JOIN?

Resposta:Com INNER JOIN, apenas os registros com valores correspondentes em ambas as tabelas são retornados. Em contraste, um LEFT JOIN retorna todos os registros da tabela da esquerda, juntamente com os registros correspondentes da tabela da direita, incluindo NULLs quando não há correspondências.

10.Pergunta

Qual é o papel dos JOINs no contexto de bancos de dados

Mais Livros Grátis
Escaneie para baixar
Ouça-o

relacionais?

Resposta:JOINs são cruciais em bancos de dados relacionais, pois conectam tabelas relacionadas usando chaves estrangeiras, permitindo que os usuários recuperem e analisem dados interconectados de forma eficiente, aumentando a utilidade e o insight proveniente do banco de dados.

Capítulo 8 | 8. JOIN| Perguntas e respostas

1.Pergunta

Qual é a função central do SQL que o distingue de outras tecnologias de dados?

Resposta:A junção de tabelas é a funcionalidade definidora do SQL, permitindo aos usuários combinar dados de várias tabelas para criar conjuntos de dados mais abrangentes e significativos.

2.Pergunta

Como as relações entre tabelas são estabelecidas no SQL?

Resposta:As relações são estabelecidas usando chaves

Mais Livros Grátis
Escaneie para baixar
Ouça-o

primárias e chaves estrangeiras, onde uma chave primária em uma tabela corresponde a uma chave estrangeira em outra, definindo como os registros em uma tabela se relacionam com os registros em outra.

3.Pergunta

Qual é a importância de uma relação um-para-muitos no design de bancos de dados?

Resposta:Uma relação um-para-muitos é crucial, pois reflete cenários do mundo real onde uma única entidade (como um cliente) pode estar vinculada a múltiplos registros em outra tabela (como pedidos), o que apoia uma organização eficiente dos dados.

4.Pergunta

O que é um INNER JOIN e como é utilizado?

Resposta:Um INNER JOIN é usado para mesclar duas tabelas, combinando registros com base em um campo comum, mostrando resultados apenas quando há registros correspondentes em ambas as tabelas.

5.Pergunta

O que acontece com os registros ao usar um INNER

Mais Livros Grátis
Escaneie para baixar
Ouça-o

JOIN se não houver correspondência em uma das tabelas?

Resposta:Os registros em uma tabela que não têm registros correspondentes na outra tabela serão excluídos dos resultados do INNER JOIN.

6.Pergunta

O que é um LEFT JOIN e como ele difere de um INNER JOIN?

Resposta:Um LEFT JOIN inclui todos os registros da tabela da esquerda e registros correspondentes da tabela da direita, com registros não correspondentes da direita resultando em nulos, enquanto um INNER JOIN inclui apenas registros correspondentes de ambas as tabelas.

7.Pergunta

Quando você utilizaria um LEFT JOIN em vez de um INNER JOIN?

Resposta:Um LEFT JOIN é usado quando é importante incluir todos os registros da tabela da esquerda, mesmo aqueles que não têm registros associados na tabela da direita,

Mais Livros Grátis
Escaneie para baixar
Ouça-o

como ao listar todos os clientes, independentemente de terem feito um pedido.

8.Pergunta

Qual é o propósito de configurar chaves estrangeiras em um banco de dados?

Resposta:As chaves estrangeiras garantem a integridade referencial entre tabelas, assegurando que as relações entre registros sejam mantidas, prevenindo a existência de registros órfãos.

9.Pergunta

Como os JOINs podem melhorar a representação de dados no SQL?

Resposta:Os JOINs permitem combinar dados de várias tabelas em uma única visão, proporcionando um contexto mais rico e informações mais detalhadas em consultas, o que torna a análise de dados mais eficaz.

10.Pergunta

Qual é a importância da normalização no design de bancos de dados?

Resposta:A normalização reduz a redundância de dados e

Mais Livros Grátis
Escaneie para baixar
Ouça-o

melhora a integridade dos dados ao organizar tabelas de forma lógica, assegurando que os dados sejam armazenados de forma eficiente e que as relações entre tabelas sejam bem definidas.

11.Pergunta

Quais são os riscos potenciais de medidas de segurança inadequadas para bancos de dados?

Resposta:Medidas de segurança inadequadas podem levar ao acesso não autorizado e a vazamentos de dados, tornando informações sensíveis vulneráveis a ataques maliciosos, como a injeção de SQL.

12.Pergunta

Quais são as perguntas críticas a considerar durante o processo de design de um banco de dados?

Resposta:As perguntas críticas de design incluem identificar requisitos empresariais, determinar tabelas e colunas necessárias, garantir normalização adequada e definir relacionamentos pai-filho.

13.Pergunta

Por que é essencial abordar tanto a população de dados

Mais Livros Grátis
Escaneie para baixar
Ouça-o

quanto a segurança ao planejar um banco de dados?
Resposta:A população de dados assegura que os dados permaneçam relevantes e precisos, enquanto a segurança protege contra vazamentos de dados, contribuindo assim para uma implementação de banco de dados bem-sucedida e confiável.

Capítulo 9 | 9. Design de Banco de Dados| Perguntas e respostas

1.Pergunta

Quais são as perguntas críticas de design que você deve considerar ao planejar um banco de dados?
Resposta:Ao planejar um banco de dados, considere as seguintes perguntas críticas de design: 1. Quais são os requisitos de negócios? 2. Quais tabelas serão necessárias para atender a esses requisitos? 3. Quais colunas cada tabela conterá? 4. Como as tabelas serão normalizadas? 5. Quais serão seus relacionamentos pai/filho? É benéfico elaborar um diagrama para visualizar esses relacionamentos.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

2.Pergunta

Por que é importante planejar a manutenção de dados no design do banco de dados?

Resposta:Planejar a manutenção de dados é crucial porque, se os dados não forem mantidos e atualizados, o design do banco de dados já falhou. Um banco de dados não mantido pode rapidamente tornar-se não confiável e pode levar a falhas em projetos devido a dados desatualizados ou incorretos.

3.Pergunta

Como você pode impor segurança em um banco de dados?

Resposta:A segurança pode ser imposta em um banco de dados considerando os seguintes fatores: 1. Determine quem deve ter acesso ao banco de dados e a quais tabelas. 2. Defina permissões de acesso somente leitura e de gravação para os usuários. 3. Estabeleça planos de backup para caso de desastre ou falha. 4. Implemente o registro de alterações nas tabelas para monitorar modificações. 5. Garanta uma

Mais Livros Grátis
Escaneie para baixar
Ouça-o

conectividade segura, especialmente se o banco de dados interagir com aplicativos web, para proteger contra vulnerabilidades como injeção SQL.

4.Pergunta

Qual é o papel das chaves primárias e estrangeiras no design do banco de dados?

Resposta:As chaves primárias fornecem uma identidade única para cada registro em uma tabela, garantindo a integridade dos dados e consultas eficientes. As chaves estrangeiras criam relacionamentos entre tabelas, ligando a chave primária em uma tabela a um registro relacionado em outra, estabelecendo assim relacionamentos pai/filho e mantendo a integridade referencial.

5.Pergunta

Qual é o propósito de criar um esquema de banco de dados?

Resposta:Um esquema de banco de dados serve como um diagrama que representa visualmente tabelas, suas colunas e os relacionamentos entre elas. Ele ajuda a entender como os

Mais Livros Grátis
Escaneie para baixar
Ouça-o

dados são estruturados e como diferentes tabelas se interconectam, facilitando o design e a consulta do banco de dados.

6.Pergunta

Como múltiplos registros podem ser inseridos em uma tabela de uma só vez?

Resposta:Múltiplos registros podem ser inseridos em uma tabela de uma só vez usando o comando INSERT com uma lista de valores separados por vírgula para cada registro. A sintaxe permite que você especifique todos os campos relevantes e seus valores correspondentes em uma única instrução, acelerando significativamente o processo de inserção de dados.

7.Pergunta

Qual é o benefício de usar a consulta SELECT antes de uma operação DELETE?

Resposta:Usar uma consulta SELECT antes de uma operação DELETE permite que você visualize quais registros serão excluídos com base em seus critérios. Essa precaução ajuda a

Mais Livros Grátis
Escaneie para baixar
Ouça-o

evitar perda acidental de dados, garantindo que você esteja ciente de quais registros serão afetados pelo seu comando DELETE.

8.Pergunta

O que faz o comando UPDATE em SQL?

Resposta:O comando UPDATE é usado para modificar registros existentes em uma tabela. Você pode alterar os valores de campos específicos com base em certas condições, alterando o estado dos dados sem precisar excluir e reinserir registros.

9.Pergunta

Por que a consideração cuidadosa do design é essencial para a resiliência do banco de dados?

Resposta:Um design cuidadoso do banco de dados é essencial para a resiliência porque garante que os dados permaneçam precisos, os relacionamentos entre os dados sejam mantidos e dados inadequados ou errôneos sejam restringidos de entrar no sistema. Um banco de dados bem projetado pode se adaptar mais facilmente a mudanças nos

Mais Livros Grátis
Escaneie para baixar
Ouça-o

requisitos de dados e consultas, tornando-se robusto a longo prazo.

10.Pergunta

O que são visões em um banco de dados e por que são úteis?

Resposta:Visões são tabelas virtuais criadas a partir de consultas SELECT que podem ser salvas no banco de dados. Elas permitem que os usuários simplifiquem consultas complexas apresentando os dados em um formato estruturado, facilitando o acesso e a manipulação dos dados sem modificar diretamente as tabelas subjacentes.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Ad

Escanear para baixar

Ler, Compartilhar, Empoderar

Conclua Seu Desafio de Leitura, Doe Livros para Crianças Africanas.

O Conceito

BOOKS FOR AFRICA × ×

Esta atividade de doação de livros está sendo realizada em conjunto com a Books For Africa.Lançamos este projeto porque compartilhamos a mesma crença que a BFA: Para muitas crianças na África, o presente de livros é verdadeiramente um presente de esperança.

A Regra

Ganhe 100 pontos → **Resgate um livro** → **Doe para a África**

Seu aprendizado não traz apenas conhecimento, mas também permite que você ganhe pontos para causas beneficentes! Para cada 100 pontos ganhos, um livro será doado para a África.

Teste gratuito com Bookey

Capítulo 10 | 10. Gerenciando Dados| Perguntas e respostas

1.Pergunta

Qual é a importância de um design de tabela bem planejado em um banco de dados?

Resposta:Um design de tabela bem planejado garante uma entrada e recuperação de dados eficientes. Permite relacionamentos eficazes entre conjuntos de dados, minimizando a redundância e facilitando as atualizações, já que a modificação de um único registro se refletirá em todo o banco de dados. Restrições adequadas evitam entradas de dados errôneas, aumentando a integridade dos dados.

2.Pergunta

Como funciona a instrução INSERT em um banco de dados relacional?

Resposta:A instrução INSERT adiciona registros às tabelas dentro de um banco de dados relacional. Você pode especificar quais campos deseja preencher, e o banco de

Mais Livros Grátis
Escaneie para baixar
Ouça-o

dados atribui valores padrão ou nulos aos outros campos. Por exemplo, se você executar 'INSERT INTO ATTENDEE (FIRST_NAME, LAST_NAME) VALUES ('Thomas', 'Nield')', cria uma nova entrada de participante com os nomes especificados.

3.Pergunta

Por que é recomendável usar instruções SELECT antes de executar comandos DELETE?

Resposta:Usar SELECT anteriormente ajuda a evitar a perda acidental de dados, mostrando quais registros serão afetados pelo comando DELETE. Dessa forma, você pode verificar se os dados corretos estão sendo alvo de exclusão, evitando erros irreversíveis.

4.Pergunta

Quais são os benefícios de realizar múltiplos INSERTs em um único comando?

Resposta:Executar múltiplos INSERTs em um único comando é muito mais eficiente, especialmente para grandes conjuntos de dados. Reduz o número de transações no banco

Mais Livros Grátis
Escaneie para baixar
Ouça-o

de dados, o que pode acelerar significativamente o tempo de processamento, em comparação com a execução de instruções INSERT individuais uma após a outra.

5.Pergunta

Qual é o papel das restrições de chave estrangeira ao inserir registros?

Resposta:As restrições de chave estrangeira mantêm a integridade referencial dentro de um banco de dados. Elas garantem que os relacionamentos entre tabelas permaneçam válidos. Por exemplo, se você tentar inserir um registro que referência um ID inexistente em outra tabela, o banco de dados rejeitará a operação, prevenindo registros órfãos.

6.Pergunta

Como o comando UPDATE pode ser aplicado condicionalmente a registros?

Resposta:O comando UPDATE pode modificar registros específicos com base em condições definidas em uma cláusula WHERE. Por exemplo, 'UPDATE ATTENDEE SET VIP = 1 WHERE ATTENDEE_ID IN (3,4)' definirá o status

Mais Livros Grátis
Escaneie para baixar
Ouça-o

VIP como verdadeiro apenas para participantes com IDs 3 e 4, permitindo alterações direcionadas ao invés de atualizações em massa.

7.Pergunta

Que precauções um desenvolvedor deve tomar ao usar o comando DROP TABLE?

Resposta:O comando DROP TABLE exclui permanentemente uma tabela e todos os seus registros, portanto, deve-se garantir que os dados não sejam mais necessários ou que um backup exista antes de executá-lo. Verificar a necessidade e as implicações da exclusão é crucial, pois não pode ser desfeita.

8.Pergunta

Como garantir uma entrada de dados eficiente para usuários que podem não conhecer SQL?

Resposta:Para acomodar usuários não familiarizados com SQL, você pode criar uma interface gráfica do usuário (GUI) que simplifique a entrada e a gestão de dados. Isso pode incluir menus suspensos, campos de entrada e formulários

Mais Livros Grátis
Escaneie para baixar
Ouça-o

que lidam com validação de dados e geração de comandos SQL nos bastidores, facilitando a interação com o banco de dados.

9.Pergunta

Quais recursos ou ações você pode buscar para expandir ainda mais suas habilidades em SQL e gestão de bancos de dados?

Resposta:Você pode se envolver em várias atividades para aprofundar seu conhecimento em SQL, como explorar tópicos avançados em livros, participar de cursos online, praticar com projetos do mundo real, utilizar fóruns comunitários como o Stack Overflow para dúvidas e aprender tecnologias complementares como Python ou R para aprimorar suas capacidades de processamento de dados.

10.Pergunta

Por que aprender várias linguagens de programação ou tecnologias pode ser benéfico para sua carreira?

Resposta:Adquirir conhecimento em várias linguagens de programação e tecnologias amplia seu conjunto de habilidades técnicas, tornando você adaptável a diferentes

Mais Livros Grátis
Escaneie para baixar
Ouça-o

funções e aumentando sua atratividade no mercado. Isso abre oportunidades em diversos campos e melhora sua capacidade de construir soluções abrangentes em múltiplas plataformas.

Capítulo 11 | 11. Indo em Frente| Perguntas e respostas

1.Pergunta

Quais são os caminhos profissionais potenciais que se pode seguir após aprender SQL e design de banco de dados?

Resposta:Após dominar SQL e design de banco de dados, você pode seguir várias carreiras, como: 1. **Analista de Dados** - Usar SQL para analisar dados e obter insights para os negócios. 2. **Administrador de Banco de Dados** - Gerenciar e manter sistemas de banco de dados. 3. **Desenvolvedor de Business Intelligence** - Utilizar SQL e ferramentas de BI para melhorar a tomada de decisões. 4. **Engenheiro de Dados** - Projetar e criar pipelines de dados escaláveis. 5. **Desenvolvedor de Software** - Construir

Mais Livros Grátis
Escaneie para baixar
Ouça-o

aplicações que interagem com bancos de dados usando linguagens de programação como Python ou Java.

2.Pergunta

Como o SQL pode ser complementado com outras linguagens de programação para aumentar a empregabilidade?

Resposta:Integrar SQL com outras linguagens de programação pode aumentar significativamente sua atratividade no mercado. Por exemplo: 1. **Python** - Ótimo para manipulação de dados e scripts. 2. **R** - Excelente para estatísticas e análise de dados. 3. **Java** - Fundamental para desenvolver aplicações e soluções em nível empresarial. 4. **C# e Swift** - Útil para construir aplicações desktop e móveis. Cada uma dessas linguagens pode melhorar a forma como você interage com bancos de dados e ampliar o escopo dos projetos que pode lidar.

3.Pergunta

Quais recursos você recomenda para aprofundar o conhecimento e as habilidades em SQL?

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Resposta:Para aprofundar seu conhecimento em SQL e nas funcionalidades de banco de dados, considere usar os seguintes recursos: 1. **Livros** como "Using SQLite" de Jay A. Kreibich para especificidades do SQLite. 2. **Cursos Online** através de plataformas como Coursera para programação em R. 3. **Sites** como W3Schools e Pluralsight para conhecimento fundamental. 4. **Recursos Comunitários** como Stack Overflow para solução de problemas e troca de conhecimentos com profissionais.

4.Pergunta

Como manter a motivação ao aprender novas tecnologias como SQL?

Resposta:Para se manter motivado ao aprender SQL ou qualquer nova tecnologia, experimente as seguintes estratégias: 1. **Defina Metas Específicas** - Divida seu aprendizado em objetivos manejáveis e específicos. 2. **Prática Regular** - Use SQL consistentemente em projetos para reforçar suas habilidades. 3. **Engaje-se com uma Comunidade** - Junte-se a fóruns ou grupos de estudo

Mais Livros Grátis
Escaneie para baixar
Ouça-o

para apoio e compartilhamento de experiências. 4. **Celebre Conquistas** - Reconheça suas realizações para incentivar o aprendizado contínuo.

5.Pergunta

Como o Google e as comunidades online podem ajudar na aprendizagem de SQL?

Resposta:O Google pode ser uma ferramenta extremamente eficaz para aprender SQL, fornecendo respostas imediatas a perguntas e problemas específicos. Comunidades online, como o Stack Overflow, permitem que você interaja com especialistas e entusiastas, que podem fornecer insights, solucionar problemas e compartilhar melhores práticas. Quando confrontado com um desafio, pesquisar em fóruns pode revelar que muitos outros enfrentaram problemas semelhantes, levando frequentemente a soluções ou alternativas.

6.Pergunta

Em que os aprendizes de SQL devem se concentrar além de dominar os comandos básicos de SQL?

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Resposta:Além de dominar os comandos básicos de SQL, os aprendizes devem se concentrar em: 1. **Entender Relacionamentos de Dados** - Familiarizar-se com o design de banco de dados e normalização. 2. **Conceitos Avançados de SQL** - Explorar subconsultas, gatilhos e funções. 3. **Ajuste de Desempenho** - Aprender técnicas de otimização para consultas e índices. 4. **Integração com Outras Tecnologias** - Compreender como o SQL interage com linguagens de programação, ferramentas de visualização de dados e análise de dados.

7.Pergunta

Qual é o papel da prática consistente para se tornar proficiente em SQL?

Resposta:A prática consistente é fundamental para dominar o SQL, pois reforça o conhecimento teórico através da aplicação no mundo real. Usar SQL regularmente para consultar bancos de dados, criar relatórios e se envolver em projetos aprimora as habilidades de resolução de problemas e aumenta a familiaridade com várias funcionalidades do SQL,

Mais Livros Grátis
Escaneie para baixar
Ouça-o

construindo, em última análise, confiança e expertise.

8.Pergunta

Como alguém pode aproveitar o conhecimento existente em SQL para transitar para campos avançados de dados como business intelligence ou engenharia de dados?

Resposta:Ter uma base sólida em SQL fornece as habilidades essenciais para transitar para campos avançados de dados. A partir daí, você deve se concentrar em: 1. **Aprender Ferramentas de BI** - Familiarizar-se com Tableau ou Power BI para visualização de dados. 2. **Extração e Transformação de Dados** - Dominar processos ETL, que envolvem consultas SQL para manipulação de dados. 3. **Entender Armazenamento de Dados** - Aprender como os bancos de dados armazenam dados de forma diferente em contextos de BI. 4. **Programação em Colaboração com SQL** - Melhorar pipelines de dados e automação através de linguagens como Python ou R.

9.Pergunta

Qual é o valor do envolvimento comunitário na aprendizagem de SQL e tecnologia em geral?

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Resposta:O envolvimento comunitário oferece um valor imenso na aprendizagem de SQL e tecnologia, proporcionando acesso a diversas bases de conhecimento. Engajar-se com comunidades permite que os aprendizes: 1. **Compartilhar Problemas e Soluções** - Colaborar em questões enfrentadas durante o aprendizado. 2. **Networking com Profissionais** - Estabelecer conexões que podem levar a mentorias ou oportunidades de emprego. 3. **Manter-se Atualizado** - Aprender sobre as últimas tendências e tecnologias na indústria. 4. **Ganhar Diferentes Perspectivas** - Compreender abordagens diversas para resolver desafios técnicos, enriquecendo seu conjunto de ferramentas de resolução de problemas.

Capítulo 12 | A. Operadores e Funções| Perguntas e respostas

1.Pergunta

Como você pode testar operadores e funções SQL sem consultar tabelas no SQLite?

Resposta:Você pode testar operadores e funções do SQLite usando consultas de expressões literais. Por

Mais Livros Grátis
Escaneie para baixar
Ouça-o

exemplo, você pode executar uma consulta simples como `SELECT 5 + 7` para ver o resultado diretamente, sem precisar de tabelas de banco de dados.

2.Pergunta

Qual é o resultado da expressão `SELECT INSTR('TONY STARK', 'TONY')`?

Resposta:O resultado é 1, indicando que a string 'TONY' é encontrada dentro de 'TONY STARK' na posição 1.

3.Pergunta

Quais são os cinco operadores matemáticos principais no SQLite?

Resposta:Os cinco operadores matemáticos principais são: + para adição, - para subtração, * para multiplicação, / para divisão e % para calcular o restante.

4.Pergunta

Como funcionam os operadores de comparação no SQLite? Você pode fornecer um exemplo?

Resposta:Os operadores de comparação retornam verdadeiro (1) ou falso (0) com base em condições avaliativas. Por

Mais Livros Grátis
Escaneie para baixar
Ouça-o

exemplo, `SELECT 5 < 10` retorna 1 (verdadeiro), enquanto `SELECT 5 > 10` retorna 0 (falso).

5.Pergunta

Qual é o propósito dos operadores lógicos em SQL?

Resposta:Os operadores lógicos são usados para combinar múltiplas expressões Booleanas, permitindo consultas mais complexas. Por exemplo, usando `AND` verifica se todas as condições dadas são verdadeiras.

6.Pergunta

Qual é a sintaxe para uma consulta SQL para encontrar o comprimento da string 'Dallas'?

Resposta:Você pode usar a função `length('Dallas')` que retorna 6 como o número de caracteres na string.

7.Pergunta

Você pode explicar como funcionam as funções agregadas no SQLite?

Resposta:As funções agregadas realizam cálculos em conjuntos de valores e retornam um único valor. Funções como `SUM()`, `AVG()`, `COUNT()`, `MIN()` e `MAX()` podem resumir dados entre linhas selecionadas.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

8.Pergunta

Por que é importante usar o formato 'YYYY-MM-DD' para datas no SQLite?

Resposta:Usar o formato 'YYYY-MM-DD' é crucial porque permite que o SQLite interprete essas strings como valores de data. Se as datas não estiverem nesse formato, o SQLite as tratará como texto simples, o que pode levar a comparações incorretas.

9.Pergunta

O que acontece quando você realiza operações usando datas no formato 'YYYY-MM-DD'? Você pode fornecer um exemplo?

Resposta:Quando você usa o formato 'YYYY-MM-DD', pode comparar datas de forma eficiente. Por exemplo, `SELECT '2015-05-14' > '2015-01-12'` retornará 1 (verdadeiro) porque 14 de maio de 2015 é posterior a 12 de janeiro de 2015.

10.Pergunta

Como as transações são tratadas no SQL e qual é o seu propósito?

Resposta:As transações agrupam múltiplas instruções SQL

Mais Livros Grátis
Escaneie para baixar
Ouça-o

para que todas possam ser bem-sucedidas ou todas falharem juntas, garantindo atomicidade. Se ocorrer um erro durante qualquer instrução em uma transação, um ROLLBACK pode desfazer todas as operações anteriores.

11.Pergunta

O que é um índice no SQLite e como ele melhora o desempenho das consultas?

Resposta:Um índice é um recurso de otimização de banco de dados que acelera consultas SELECT ao fornecer um mecanismo de busca rápido. No entanto, ele pode desacelerar operações de INSERT, UPDATE e DELETE, pois o índice deve ser atualizado.

12.Pergunta

O que acontece se você encontrar um erro durante uma transação?

Resposta:Se ocorrer um erro durante uma transação, um comando ROLLBACK pode ser emitido para desfazer todas as alterações feitas durante aquela transação, mantendo a integridade do banco de dados.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

13.Pergunta

Como você cria um índice em uma coluna no SQLite?

Resposta:Para criar um índice, você pode executar uma consulta como `CREATE INDEX index_name ON table_name(column_name);`. Isso melhorará o desempenho das consultas que filtram por essa coluna.

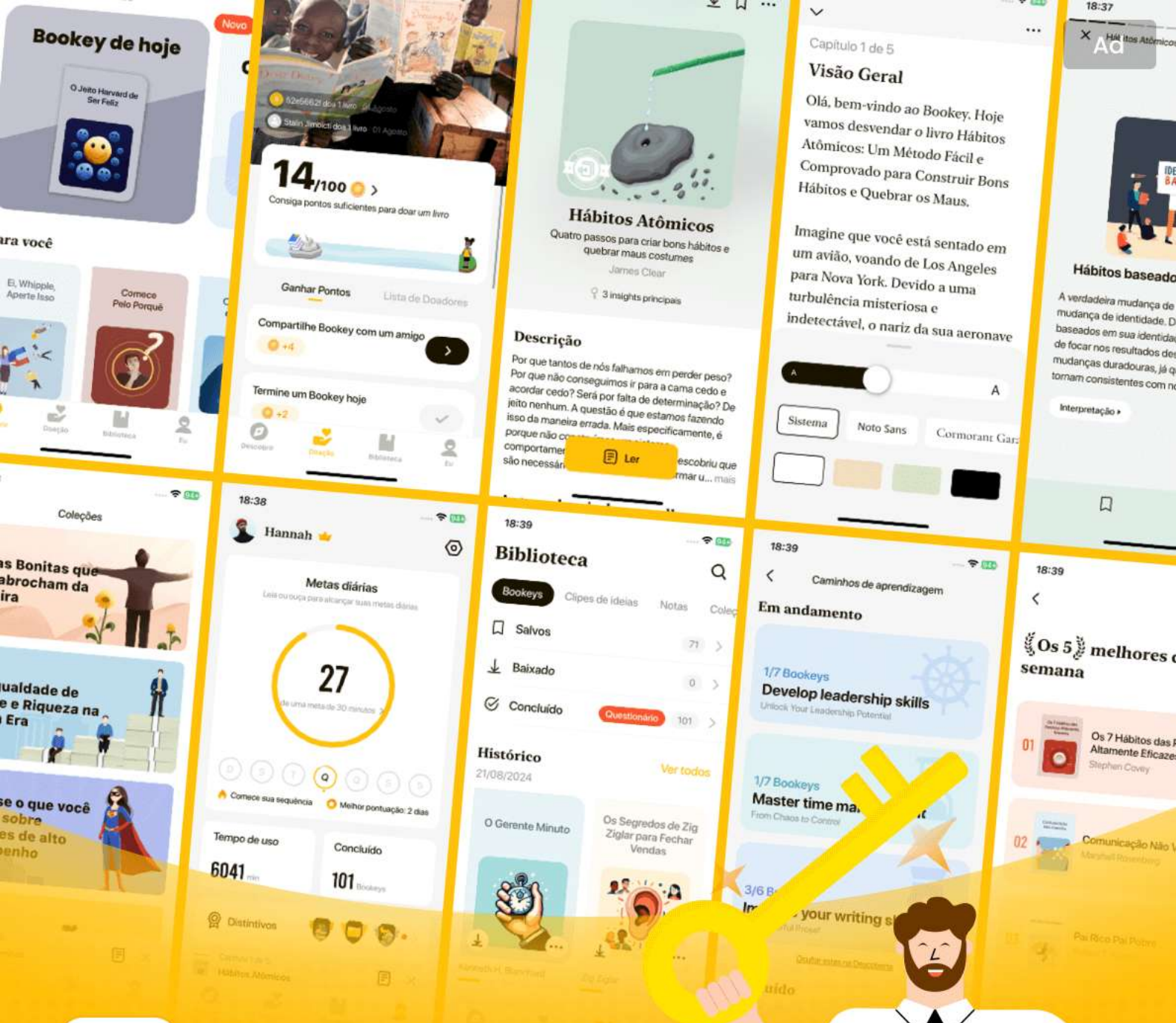
14.Pergunta

Qual é a diferença entre um UNION e um UNION ALL em SQL?

Resposta:`UNION` combina os resultados de duas ou mais consultas SELECT e remove linhas duplicadas, enquanto `UNION ALL` combina os resultados, mas mantém todas as duplicatas.

Mais Livros Grátis

Escaneie para baixar
Ouça-o



As melhores ideias do mundo desbloqueiam seu potencial

Essai gratuit avec Bookey

Descarregar na App Store

OBTER NO Google Play

Escanear para baixar

Capítulo 13 | B. Tópicos Suplementares| Perguntas e respostas

1.Pergunta

Quais estratégias podem ajudar a melhorar o desempenho de consultas em bancos de dados à medida que crescem?

Resposta:Para melhorar o desempenho em um banco de dados em crescimento, você pode implementar índices, que permitem buscas mais rápidas, semelhante a um índice em um livro. No entanto, lembre-se de que, embora os índices acelerem consultas SELECT, eles podem desacelerar operações de INSERT, UPDATE e DELETE. É melhor criar índices após identificar problemas de desempenho, em vez de durante o design inicial.

2.Pergunta

Qual é o propósito de usar transações em SQL e você pode fornecer um exemplo?

Resposta:Transações garantem que múltiplas operações SQL sejam executadas como uma única unidade de trabalho,

Mais Livros Grátis
Escaneie para baixar
Ouça-o

mantendo a integridade dos dados. Por exemplo, em uma transação financeira para transferir dinheiro de uma conta bancária para outra, você utilizaria uma transação para garantir que tanto as operações de débito quanto de crédito sejam bem-sucedidas ou falhem juntas. Se uma operação falhar, você pode reverter ambas, prevenindo a corrupção de dados.

3.Pergunta

Como a atomicidade em transações protege a integridade dos dados durante operações complexas?

Resposta:A atomicidade garante que uma série de operações sejam tratadas como uma única unidade. Por exemplo, ao transferir fundos entre duas contas, se uma parte da operação falhar (como esquecer um ID de conta), a atomicidade garante que a operação possa ser revertida completamente, evitando possíveis perdas de dados ou discrepâncias.

4.Pergunta

Por que os índices devem ser usados com cautela? Quais podem ser as desvantagens do uso excessivo deles?

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Resposta:Embora os índices melhorem o desempenho das consultas, eles podem levar a um aumento no tamanho do banco de dados e desacelerar operações de modificação de dados, como INSERT, UPDATE e DELETE. O uso excessivo de índices, especialmente em tabelas pequenas ou tabelas que são modificadas com frequência, pode degradar o desempenho geral do sistema em vez de melhorá-lo.

5.Pergunta

Quais tópicos adicionais de SQL os iniciantes devem explorar após este livro fundamental?

Resposta:Após dominar os conceitos fundamentais de SQL neste livro, os iniciantes podem explorar tópicos como UNION e UNION ALL para combinar resultados de múltiplas consultas, subconsultas, triggers para ações automatizadas, expressões regulares para correspondência avançada de texto e administração de banco de dados para otimizar grandes ambientes corporativos.

6.Pergunta

Você pode explicar a diferença no impacto de desempenho entre o uso de índices em tabelas grandes e pequenas?

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Resposta:Em tabelas grandes, os índices podem aumentar significativamente o desempenho ao acelerar as consultas SELECT. No entanto, em tabelas pequenas—onde o impacto no desempenho é mínimo—os índices podem introduzir uma sobrecarga desnecessária e, na verdade, podem desacelerar o desempenho geral devido à manutenção adicional durante as modificações de dados.

7.Pergunta

Qual é a importância de usar um índice UNIQUE em SQL?

Resposta:Um índice UNIQUE garante que todos os valores na coluna indexada sejam distintos. Isso pode otimizar o desempenho das consultas e as verificações de integridade, já que o banco de dados impedirá entradas duplicadas, tornando as buscas mais rápidas e mantendo a consistência dos dados.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Ad

Escanear para baixar

Experimente o aplicativo Bookey para ler mais de 1000 resumos dos melhores livros do mundo

Desbloqueie 1000+ títulos, 80+ tópicos

Novos títulos adicionados toda semana

duct & Brand | Liderança & Colaboração | Gerenciamento de Tempo | Relacionamento & Comunica

Estratégia de Negócios | Criatividade | Memórias | Conheça a Si Mesmo | Psicologia Po

mpreendedorismo | História Mundial | Comunicação entre Pais e Filhos | Autocuidado | Mi

Visões dos melhores livros do mundo

Teste gratuito com Bookey

Introduçao a Linguagem SQL Quiz e teste

Ver a resposta correta no site do Bookey

Capítulo 1 | 1. Por Que Aprender SQL?| Quiz e teste

1. SQL é útil apenas para profissionais de TI e não tem aplicação para pessoas de negócios ou gerentes.
2. Bancos de dados relacionais organizam os dados em tabelas com relacionamentos para reduzir a redundância e melhorar a manutenibilidade.
3. Bancos de dados leves como MySQL são ideais para projetos individuais ou de pequenas equipes.

Capítulo 2 | 2. Bancos de Dados| Quiz e teste

1. Um banco de dados é qualquer coleção estruturada de dados, que inclui planilhas e arquivos de texto simples, juntamente com sistemas de gerenciamento de banco de dados relacionais (RDBMS).
2. A normalização em bancos de dados visa aumentar a

Mais Livros Grátis
Escaneie para baixar
Ouça-o

redundância, armazenando todas as informações em uma única tabela para simplificar a gestão de dados.

3. SQL permanece amplamente consistente em diferentes plataformas de banco de dados, tornando o conhecimento sobre ele transferível, mesmo que haja nuances nos sistemas individuais.

Capítulo 3 | 3. SQLite| Quiz e teste

1. O SQLite foi projetado principalmente para ambientes multiusuário e pode lidar com acessos simultâneos de forma eficaz.
2. O SQLiteStudio é recomendado como um editor de SQL amigável que requer instalação para gerenciar bancos de dados SQLite.
3. A instrução SELECT em SQL é usada para recuperar dados de uma ou mais tabelas e pode incluir cálculos ou expressões.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Escanear para baixar

Baixe o app Bookey para desfrutar

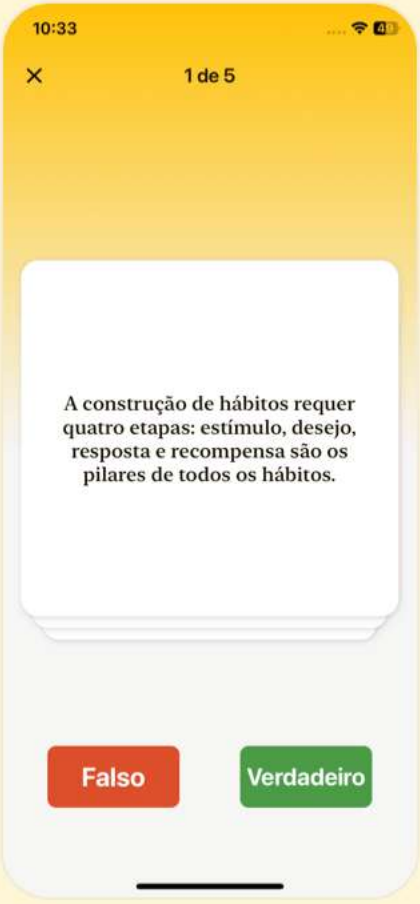
Mais de 1000 resumos de livros com quizzes

Teste grátis disponível!

Descarregar na App Store

OBTER NO Google Play

18:37
Hábitos Atômicos
Quatro passos para criar bons hábitos e quebrar maus costumes
James Clear
3 insights principais
Descrição
Por que tantos de nós falhamos em perder peso? Por que não conseguimos ir para a cama cedo e acordar cedo? Será por falta de determinação? De jeito nenhum. A questão é que estamos fazendo isso da maneira errada. Mais especificamente, é porque não
Ler
mais


10:33
1 de 5
A construção de hábitos requer quatro etapas: estímulo, desejo, resposta e recompensa são os pilares de todos os hábitos.
Falso
Verdadeiro

10:33
5 de 5
A Regra dos Dois Minutos é uma maneira rápida de acabar com a procrastinação, mas só funciona por dois minutos e faz pouco para construir hábitos de longo prazo.
Falso
Resposta correta
Uma vez que você aprendeu a cuidar da semente de cada hábito, os primeiros dois minutos são apenas o início das questões formais. Com o tempo, você esquecerá o limite de tempo de dois minutos e melhorará na construção do hábito.
Continuar

Capítulo 4 | 4. SELECT| Quiz e teste

1. A instrução SELECT pode ser usada apenas para exibir dados, não para transformações ou cálculos.
2. Em SQL, ao filtrar com a cláusula WHERE, os valores de texto devem estar entre aspas duplas.
3. Usar aliases em instruções SELECT pode ajudar a dar nomes personalizados a colunas calculadas ou colunas existentes.

Capítulo 5 | 5. WHERE| Quiz e teste

1. A cláusula WHERE é utilizada em SQL para filtrar registros com base em critérios especificados.
2. A cláusula WHERE pode comparar valores NULL usando operadores padrão.
3. A instrução ORDER BY classifica os resultados de uma consulta em ordem crescente por padrão.

Capítulo 6 | 6. GROUP BY e ORDER BY| Quiz e teste

1. SQL permite agrupar dados sem usar colunas

Mais Livros Grátis
Escaneie para baixar
Ouça-o

especificadas.

2. A ordem de classificação padrão em SQL é decrescente ao usar ORDER BY.
3. A cláusula HAVING é usada para filtrar resultados agregados em consultas SQL.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Escanear para baixar

Baixe o app Bookey para desfrutar

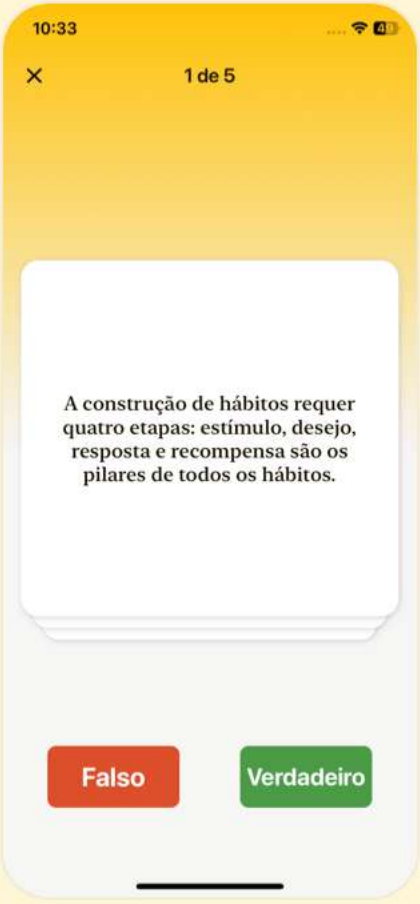
Mais de 1000 resumos de livros com quizzes

Teste grátis disponível!

Descarregar na App Store

OBTER NO Google Play

18:37
Hábitos Atômicos
Quatro passos para criar bons hábitos e quebrar maus costumes
James Clear
3 insights principais
Descrição
Por que tantos de nós falhamos em perder peso? Por que não conseguimos ir para a cama cedo e acordar cedo? Será por falta de determinação? De jeito nenhum. A questão é que estamos fazendo isso da maneira errada. Mais especificamente, é porque não
Ler
mais


10:33
1 de 5
A construção de hábitos requer quatro etapas: estímulo, desejo, resposta e recompensa são os pilares de todos os hábitos.
Falso
Verdadeiro

10:33
5 de 5
A Regra dos Dois Minutos é uma maneira rápida de acabar com a procrastinação, mas só funciona por dois minutos e faz pouco para construir hábitos de longo prazo.
Falso
Resposta correta
Uma vez que você aprendeu a cuidar da semente de cada hábito, os primeiros dois minutos são apenas o início das questões formais. Com o tempo, você esquecerá o limite de tempo de dois minutos e melhorará na construção do hábito.
Continuar

Capítulo 7 | 7. Instruções CASE| Quiz e teste

1. Uma declaração CASE em SQL começa com 'CASE' e termina com 'END'.
2. LEFT JOIN inclui apenas registros onde há correspondência em ambas as tabelas, desconsiderando o restante.
3. Agrupar por condições em SQL permite transformações e agregações de dados aprimoradas.

Capítulo 8 | 8. JOIN| Quiz e teste

1. Uma operação INNER JOIN inclui registros de ambas as tabelas, mesmo que não haja um registro correspondente em uma das tabelas.
2. LEFT JOIN é usado para incluir todos os registros da tabela da esquerda, mesmo que não haja correspondências na tabela da direita.
3. Ter um esquema de banco de dados bem projetado é desnecessário desde que os dados sejam populados corretamente.

Capítulo 9 | 9. Design de Banco de Dados| Quiz e

Mais Livros Grátis
Escaneie para baixar
Ouça-o

teste

1. Uma chave primária é usada para identificar registros de forma única em uma tabela de banco de dados.
2. Chaves estrangeiras podem existir sem chaves primárias correspondentes em tabelas pai.
3. Criar visualizações em um banco de dados permite um acesso mais fácil a consultas frequentemente usadas.

Mais Livros Grátis

Escaneie para baixar
Ouça-o

Escanear para baixar

Baixe o app Bookey para desfrutar

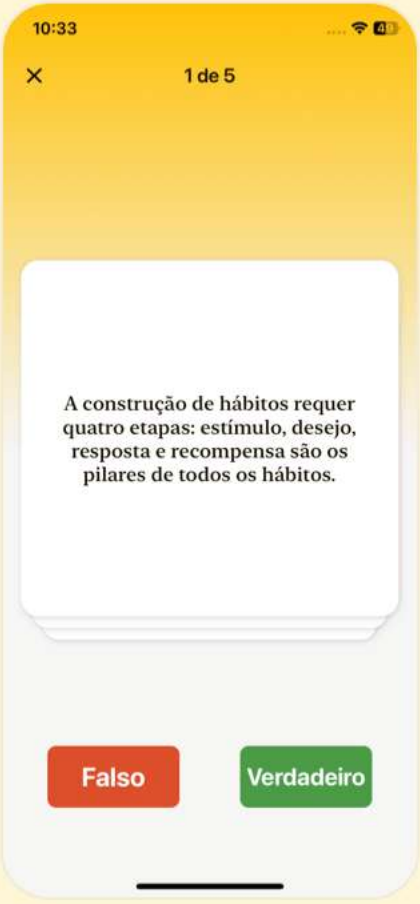
Mais de 1000 resumos de livros com quizzes

Teste grátis disponível!

Descarregar na App Store

OBTER NO Google Play

18:37
Hábitos Atômicos
Quatro passos para criar bons hábitos e quebrar maus costumes
James Clear
3 insights principais
Descrição
Por que tantos de nós falhamos em perder peso? Por que não conseguimos ir para a cama cedo e acordar cedo? Será por falta de determinação? De jeito nenhum. A questão é que estamos fazendo isso da maneira errada. Mais especificamente, é porque não
Ler
mais


10:33
1 de 5
A construção de hábitos requer quatro etapas: estímulo, desejo, resposta e recompensa são os pilares de todos os hábitos.
Falso
Verdadeiro

10:33
5 de 5
A Regra dos Dois Minutos é uma maneira rápida de acabar com a procrastinação, mas só funciona por dois minutos e faz pouco para construir hábitos de longo prazo.
Falso
Resposta correta
Uma vez que você aprendeu a cuidar da semente de cada hábito, os primeiros dois minutos são apenas o início das questões formais. Com o tempo, você esquecerá o limite de tempo de dois minutos e melhorará na construção do hábito.
Continuar

Capítulo 10 | 10. Gerenciando Dados| Quiz e teste

1. A instrução INSERT pode ser usada para adicionar novos registros a uma tabela de banco de dados, e permite que você preencha apenas campos específicos, deixando outros como NULL ou valores padrão.
2. A instrução DELETE só pode remover registros que atendem a condições especificadas e não pode deletar todos os registros de uma tabela de uma vez.
3. TRUNCATE TABLE é um comando que redefine campos de auto incremento e remove todos os registros de uma tabela, e é suportado por todos os sistemas SQL, incluindo SQLite.

Capítulo 11 | 11. Indo em Frente| Quiz e teste

1. SQL é útil apenas para quem busca uma carreira em TI.
2. Aprender Python pode aumentar sua adaptabilidade e atratividade no mercado em conjunto com SQL.
3. A função DATE() no SQLite pode ser usada para manipular

Mais Livros Grátis
Escaneie para baixar
Ouça-o

datas armazenadas no formato AAAA-DD-MM.

Capítulo 12 | A. Operadores e Funções| Quiz e teste

1. O SQLite inclui um conjunto específico de operadores matemáticos, como adição, subtração, multiplicação, divisão e resto.
2. O operador 'LIKE' no SQLite é usado para correspondência exata de strings sem quaisquer padrões.
3. Transações no SQLite garantem que todas as operações dentro do bloco de transação sejam executadas ou nenhuma delas, mantendo a integridade dos dados.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Escanear para baixar

Baixe o app Bookey para desfrutar

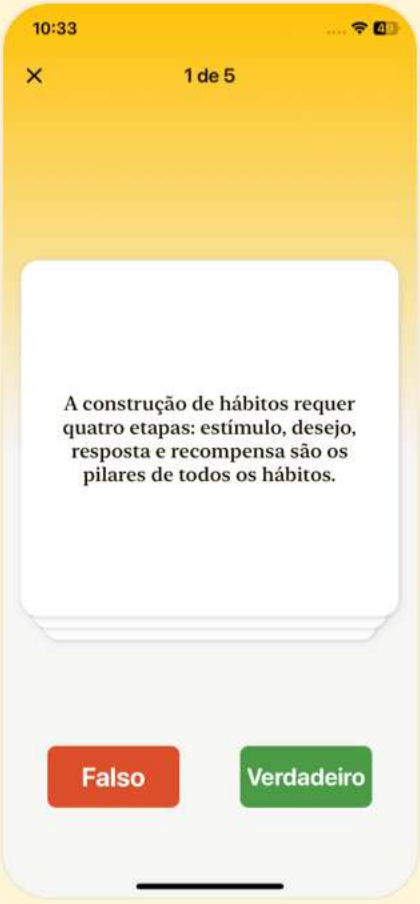
Mais de 1000 resumos de livros com quizzes

Teste grátis disponível!

Descarregar na App Store

OBTER NO Google Play

18:37
Hábitos Atômicos
Quatro passos para criar bons hábitos e quebrar maus costumes
James Clear
3 insights principais
Descrição
Por que tantos de nós falhamos em perder peso? Por que não conseguimos ir para a cama cedo e acordar cedo? Será por falta de determinação? De jeito nenhum. A questão é que estamos fazendo isso da maneira errada. Mais especificamente, é porque não
Ler
mais


10:33
1 de 5
A construção de hábitos requer quatro etapas: estímulo, desejo, resposta e recompensa são os pilares de todos os hábitos.
Falso
Verdadeiro

10:33
5 de 5
A Regra dos Dois Minutos é uma maneira rápida de acabar com a procrastinação, mas só funciona por dois minutos e faz pouco para construir hábitos de longo prazo.
Falso
Resposta correta
Uma vez que você aprendeu a cuidar da semente de cada hábito, os primeiros dois minutos são apenas o início das questões formais. Com o tempo, você esquecerá o limite de tempo de dois minutos e melhorará na construção do hábito.
Continuar

Capítulo 13 | B. Tópicos Suplementares| Quiz e teste

1. Os índices podem melhorar o desempenho de consultas SELECT em grandes conjuntos de dados, mas podem diminuir a velocidade das operações de INSERÇÃO, ATUALIZAÇÃO e EXCLUSÃO.
2. Os gatilhos são usados para executar operações em lotes para comandos de INSERÇÃO, ATUALIZAÇÃO e EXCLUSÃO apenas durante o horário comercial.
3. As transações são usadas para executar instruções SQL em operações independentes separadas para garantir a integridade dos dados.

Mais Livros Grátis
Escaneie para baixar
Ouça-o

Escanear para baixar

Baixe o app Bookey para desfrutar

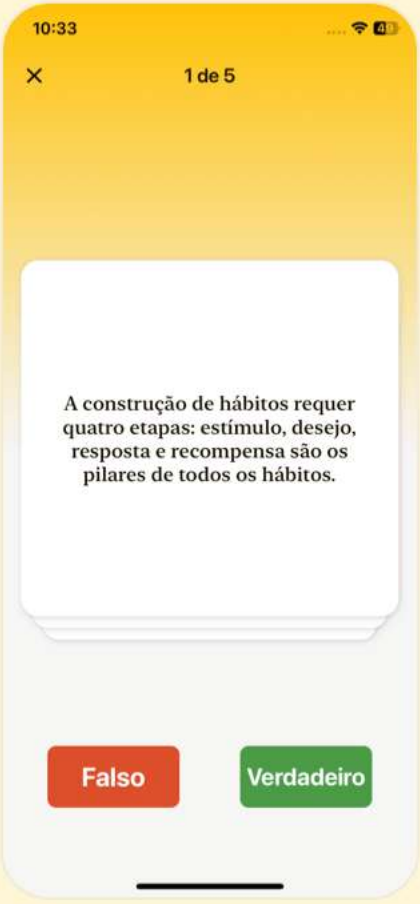
Mais de 1000 resumos de livros com quizzes

Teste grátis disponível!

Descarregar na App Store

OBTER NO Google Play

18:37
Hábitos Atômicos
Quatro passos para criar bons hábitos e quebrar maus costumes
James Clear
3 insights principais
Descrição
Por que tantos de nós falhamos em perder peso? Por que não conseguimos ir para a cama cedo e acordar cedo? Será por falta de determinação? De jeito nenhum. A questão é que estamos fazendo isso da maneira errada. Mais especificamente, é porque não
Ler
mais


10:33
1 de 5
A construção de hábitos requer quatro etapas: estímulo, desejo, resposta e recompensa são os pilares de todos os hábitos.
Falso
Verdadeiro

10:33
5 de 5
A Regra dos Dois Minutos é uma maneira rápida de acabar com a procrastinação, mas só funciona por dois minutos e faz pouco para construir hábitos de longo prazo.
Falso
Resposta correta
Uma vez que você aprendeu a cuidar da semente de cada hábito, os primeiros dois minutos são apenas o início das questões formais. Com o tempo, você esquecerá o limite de tempo de dois minutos e melhorará na construção do hábito.
Continuar